I/O NEWS

The XXU Letters
T/H/C Maillist/Labels
1987 IACU Survey & SIR

THE OFFICIAL PUBLICATION OF THE INTERNATIONAL ASSOCIATION OF CROMEMCO USERS

Volume Six, Number One

Single Copy Price $10.00

New Cromemco System Has Removable Hard Disk

The history of Cromemco is one of technical innovation. Now, in the latest design from the Cromemco wizards, comes a system with up to 760 megabytes of removable hard disk storage!

The new system is called the **System 250**, and it is the latest in a proud lineage of "black boxes" that began with the Cromemco Z-2 computer over a decade ago. The System 250 uses Cromemco's 32-bit XXU processor technology, and has a full 21 card slots for system expansion. The system can be configured with up to 16 megabytes of RAM memory and can support up to 65 serial channels. It is also an ideal platform for Cromemco's S-series graphics products.

The most unusual feature of the System 250 is its removable hard disk. In another technical first, Cromemco has developed what it calls a "MegaSafe™" cartridge for the System 250. This cartridge actually contains a fully shock-mounted 5-inch, high capacity Winchester disk drive. The cartridge can easily be inserted in, or removed from, the System 250 front panel. When installed, the cartridge is secured by a screw-type lock.

The storage capacity available in the MegaSafe™ cartridge is truly phenomenal. The cartridge is offered in three different capacities: 190 megabytes, 380 megabytes, or 760 megabytes. (The 190 megabyte and 380 megabyte cartridge are available now; the 760 megabyte cartridge will be available in the Fall). The MegaSafe™ storage capacities are particularly awesome when you consider that the world's first computer to include an internal hard disk drive (the Cromemco Z-2H) had just 11 megabytes of capacity!

In addition to the front panel mounted hard disk, the System 250 includes a 5-inch floppy drive and a 60 megabyte cartridge tape unit mounted on the front panel. The cartridge tape unit offers nearly three times the

Continued on page 28

Cromemco vs. The World (or most of it, anyway)

by Dr. Stephen Huber

In a previous issue of *I/O NEWS* (Vol. V, #5) I presented a suite of applications-specific benchmark programs and applied them to the DPU and DPU plus MAXIMIZER systems. These benchmarks tested both the single and multi-user environments. Since then, they were used to compare selected Cromemco systems with other well known micro, mini, and mainframe computer systems.

Results of these comparisons are shown in **Table 1** where execution in

Continued on page 35

Z80 Slave Processor for Cromix

by Charley Dobson

Following the recent announcement of Systems Atlanta's new Z80H slave processor with supporting software for Cromemco CROMIX systems (*I/O NEWS, Vol. 4, No. 6*), many users share our excitement concerning this new product and have requested further information. What capabilities does it offer? Just how much expansibility does it permit? How much more speed and processing power does it afford?

Cromemco's CROMIX operating system is one of the finest in the industry and much of the 68000 software is excellent. However, many CROMIX users still depend heavily upon CP/M (Z80) based software. Cromemco made it possible to run CP/M software under

Continued on page 32

Finally!

A 68000 CROMIX™ SPREADSHEET

68Kalc™ is the 68000 Cromix spreadsheet you've needed for so long.

68Kalc runs directly on the 68000 in your Cromemco computer, taking full advantage of its size and speed. 68Kalc runs on any terminal supported by Cromix termcaps, so you can use it with most popular terminals.

68Kalc is easy to learn. Its commands are a subset of Lotus 1-2-3's. Its command keystrokes mimic 1-2-3's. And help is instantly available inside the spreadsheet.

Order your first copy of 68Kalc now. Try it for thirty days. If you can imagine getting your job done without it after that, return it for a full refund.

ORDER NOW! Your first copy of 68Kalc is just $495.00 (U.S.). To order fill out the response card or call toll-free **1-800-227-3094** and have your VISA, MasterCard or company P.O. ready.

Inquiries and orders can also be addressed to: Warecraft, 501 North 36th Street, #138, Seattle, Washington 98103 USA, (206) 527-5403

Phone Orders
1-800-227-3094

Dealer Inquiries Invited.

Warecraft

68Kalc is a trademark of Warecraft.
Cromix is a registered trademark of Cromemco, Inc.
1-2-3 is a registered trademark of Lotus Development Corp.

table of contents

May/June, 1987 Volume Six, Number One

I/O NEWS

The Official Publication of The International Association of Cromemco Users is available through membership in the association. Editorial and advertising policies are designed for the enlightenment of the members in regard to new uses for, and developments of, Cromemco products and other products compatible with Cromemco systems.

I/O NEWS [ISSN 0274-9998] is published bi-monthly by The International Association of Cromemco Users (a California corporation), 24843 Del Prado, Suite 473, Dana Point, CA 92629-2852. General Offices are at 34021 Granada #B, Dana Point, CA 92629. Telephone: (714) 661-9764. Second Class Postage Paid at Dana Point and Santa Ana, CA.

POSTMASTER: Send address changes to I/O NEWS, 24843 Del Prado, Suite 473, Dana Point, CA 92629-2852.

Subscriptions to I/O NEWS are entered with membership in The IACU. Yearly memberships may be purchased for $42 (U.S. delivery address), $49 (delivery address in Canada or Mexico), and $60 (other international delivery addresses). Contact IACU for multi-year membership rates. Back issues of I/O NEWS are available for $10.00 per issue. Please note: all prices are in U.S. dollars. Return postage must be included with all manuscripts and photos submitted if they are to be returned. The IACU and I/O NEWS accept no responsibility for the return of unsolicited materials.

All rights in letters sent to IACU and I/O NEWS will be treated as unconditionally assigned for publication and copyright to comment editorially and to edit.

Copyright © 1987 by The International Association of Cromemco Users. All rights reserved. Nothing may be reprinted in whole or in part without written permission of the publisher.

William E. Jaenicke
Editor and Publisher

Lisa B. Jaenicke
Business Manager

Art Direction: Howard Millman
Printing: Hallmark Litho
Typography: Western Outdoors

COVER FEATURES

8 The XXU Letters
12 Managing Your Mail Lists
15 1987 IACU Survey & SIR

ARTICLES & FEATURES

Cover New Cromemco System Has Removable Hard Disk
Cover Cromemco vs. The World (or most of it, anyway)
Cover Z80 Slave Processor for Cromix
40 Winning at Copyrights

DEPARTMENTS

4 INPUT
6 OUTPUT
17 NEW PRODUCTS
19 SOFT TOOLS
23 32K CLASSROOM
25 C-10 ENCOUNTERS
27 INSIDE CROMIX
29 USER NOTES
36 BITS & BYTES

INPUT...

Editor:

I ran out of disk space with my 20 Mbyte drive, so I bought a Micropolis 1325 hard disk drive (which can be bought at a very attractive price, since the introduction of the 1350 series). Here are the initialization parameters that work flawlessly along with the partition layout that I use:

Disk Parameters

Number of heads	= 8
Number of cylinders	= 1024
Number of alternate tracks	= 40
Start of write precompensation	= 1024
Location of alternate tracks	= 512
Number of sectors per track	= 20
Bytes per sector	= 512
Location of alternate track table	= 9792
Location of partition table	= 9728
Starting cylinder of disk	= 1
Disk label	= CSTD

Partition #	Starting Cylinder	Size (Kbytes)
0	1	23440
1	294	11120
2	433	23440
3	726	23440

I have enclosed a blank SIR (System Improvement Request) form along with two suggestions. I can think of many more improvements to Cromix-Plus like adding numerical shell variables, or having string manipulation primitives for the string shell variables. By the way, I deserve no credit for the idea of SIR'S which are routinely managed by DECUS (DEC User's Society), for example.

Alberic Muller
Switzerland

Editor's Note: *The SIR examples submitted by Mr. Muller are presented below. In addition, a blank SIR Submission Form has been designed and appears on the back side of the 1987 IACU General Survey (page 15). See OUTPUT for details.*

SYSTEM IMPROVEMENT REQUEST SUBMISSION FORM

Submittor : Alberic Muller
Address : Rue du Jura 12
2525 Le Landeron
Switzerland
Phone : +41 38 51 41 47

Page 1 of 1

How to write a SIR:
Describe the capability you would like to see available on Cromemco Systems. Be as specific as possible. Please do not assume everybody knows how it's done on the XYZ system. Justify why the capability which you wish would be useful and give an appropriate example of its use. If you wish, suggest a possible implementation of your request.

Abstract: While running a Cromix-Plus shell command file, Control-C will abort the program executing as well as the shell under which it was started (if "abortenable"). It would be nice to control the execution of the shell and the program separately, so that the shell could continue, e.g., do some housekeeping.

Description and examples:
A new tty mode could control the shell behavior in case of a Control-C abort. New shell commands could be made to control the flow of execution within it:

1. Testing if the program last executed was aborted by Control-C.
2. Diverting the flow to some declared location (as can be done in most BASIC interpreters).

Example using method 1.

```
mode -shellabort          % disable shell abort on Control-C
maklink *.com .           % link pascal compiler
maklink *.ovr .           % and its overlays
  ...
mtplus pasprog            % compile a program
if -abort goto rem_links  % check if compilation was aborted
  ...
%rem_links
delete *.com *.ovr        % remove links from current dir.
mode shellabort           % establish default behavior again
echo "Task interrupted"   % warn user
exit
```

Example using method 2.

```
on shellabort goto rem_links
  ...
%rem_links
delete *.com *.ovr
on shellabort stop
echo "Task interrupted"
exit
```

SYSTEM IMPROVEMENT REQUEST SUBMISSION FORM

...

Abstract: Device modes are often manipulated in Cromix-Plus. It would be useful to be able to save some settings and restore them with a simple command, since it is not very practical to probe them one by one, and to restore them subsequently.

Description and examples:

```
mode save                    %save all modes
mode -ab -w 0 -wrap -l 0  % set a few modes
  ...
mode -save                   % Restore previous modes
```

--- *** ---

Editor:

I would like you to publish a portion of this letter in INPUT to see if other Cromemco users will contribute suggestions as to how I should go about particular aspects of what I want to do which they have met and solved, and also if they will contribute to I/O NEWS any solutions in both hardware wiring and software programming for the range of I/O boards available to use in Cromemco systems.

I wish to use the Cromix-Plus operating system on my hardware to combine ecological simulation modeling applied to Soil Moisture—Plant Growth—Management of Sheep & Cattle rearing and harvesting wool and meat, with real-time input of climatic variables, some recorded continuously and others at intervals; some input to the simulation model frequently and others input at intervals to correct the simulation model's estimate of parameters before they become wildly inaccurate, as well as the keyboard input of live-stock management decisions as they occur.

Sensors for:

Solar Radiation; Air Temperature (Wet and Dry); Soil Temperature; Wind Direction, Run, and Speed. Values for Rainfall and Evaporation to be manually entered.

Hardware (can be split between two computers if necessary):

ZPU, 16FDC, WDI-II, CTI, Dazzler. 64KZ-II, D+7A, 4PIO, C-1. 3 x 256KZ, TUART, PRI, GPIB. Printer: micro84P; Card Reader: Micromark II. 5 & 10 Mb IMI hard disk drives, CTD. DPU, 64FDC, WDI-II, 5¼" & 8" floppy disk drives. C-10 terminal.

Software:

CDOS, SBASIC, CROMIX, CROMIX-PLUS, CCC-D, and FORTRAN 77.

1. What Cromix-Plus functions could I cut out to save operating system memory demand assuming I run as a single user, multi-tasking system with tasking as follows:

i) Sensor reading & processing task (background).
ii) If two computers used, then computer to computer communication task (background).
iii) Simulation model execution (background).
iv) Display of simulation model results and prompt for keyboard input.
v) Development of amendments to simulation routines, or,
vi) Further directed processing of simulation mode results.

2. Should I split the work between the two computers and if so, how and which operating system should be used with the ZPU?

3. Can anyone contribute to I/O NEWS wiring diagrams for connecting particular sensors to particular boards (including 12ADC and 12DAC or others that can be used in Cromemco systems) and the corresponding software to obtain and correct sensor readings (e.g. for calibration and temperature)?

4. What data base could be used with Fortran 77, C and Sbasic or Ratfor (Fortran IV) and which combination should be preferred?

5. Other language combinations I should consider, bearing in mind Landsat and Topographical mapping will be used in pixel form and color presentation is a future development.

I must emphasize that I am an individual doing this work as a hobby in what little spare time I have from working on a grazing property and my resources are small.

Gabriel J. Harris, B.Sc., B.E.
Tregonning Hill
Stuart Town
New South Wales
Australia 2820

Editor's Note: *I sincerely hope that one or more of you reading this can be of some help to Mr. Harris. If you write him directly, please also send a copy of your suggestions to I/O NEWS.*

OUTPUT...

This issue represents a milestone of sorts. It's the first of **Volume VI**, which makes it the 31st edition of *I/O NEWS*. And as with an anniversary, it's a good time to take a look at where we're at.

But where we're at is (or should be) largely determined by where you're at. We are an association of Cromemco users after all. So in order to get more in touch with you, we've prepared a little questionnaire.

The 1987 IACU General Survey

The last survey we made was in 1982 and things have changed a mite since then. This survey form is the quickest way to help get us up to date. You'll find it on page XX. Don't be intimidated by the number of check boxes—they'll give us a more complete picture. The better we know the details of your system use, the better we'll be able to serve you. When responding to the "fill in the blank" questions, please be candid. Your suggestions and criticisms will be used to improve I/O NEWS and the IACU.

On the back side of the survey is a SIRvey (System Improvement Request form). Through it you can voice your suggestions as to how you feel Cromemco can enhance their products.

Lisa Jaenicke

Why not take a few minutes right now and complete the survey. When you have, cut the page out of the magazine and mail it to us. The information collected thru both surveys will be compiled and the results shared in *I/O NEWS*.

And when you do mail it, please be sure to send it to our new mailing address, which is:

The IACU - I/O NEWS
24843 Del Prado, Suite 473
Dana Point, CA 92629-2852
USA

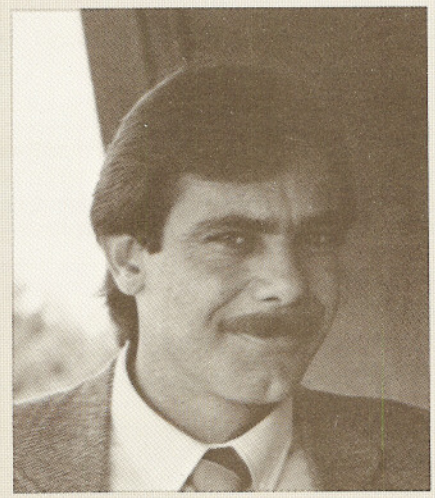
Special Inserts

You may have already noticed something a little different about this issue—a blow-in card from the folks at **WARECRAFT** and a stitched-in card from our friends at **S-100 JOURNAL**. They both have something special to offer.

WARECRAFT has great news for Cromix users—a full function 68000 spreadsheet program, **68Kalc™**. We can finally bid farewell to the old Z80 one that we've been compelled to continue using! Check out their ad on the inside front cover.

And if you haven't yet seen a copy of **S-100 JOURNAL**, now's your chance. Just complete and return the reply card for a free issue. We think you'll like it.

When you finish filling out the survey, and have ordered your copy of **68Kalc™** and **S-100 JOURNAL**, you'll be ready to sit back and dig into this issue's delights. On the front cover, Cromemco showcases their latest development, dubbed the **System 250**, an XXU-based system with a removable hard disk. But the real star of this issue is the XXU itself.



Bill Jaenicke

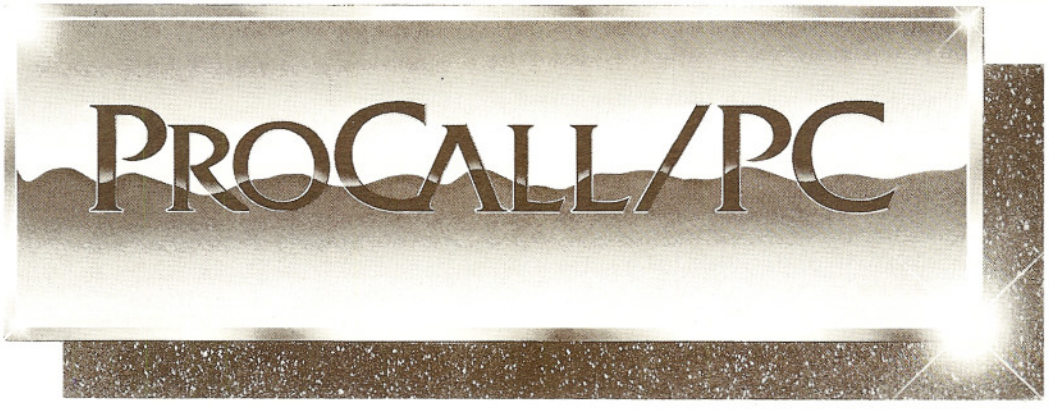
Between Stephen Huber's look at "Cromemco vs. the World" and the "XXU Letters" it becomes clear that the performance of Cromemco's 32-bit XXU processor board is nothing short of phenomenal. If these two informative articles give you the itch for an XXU, Cromemco's special summer pricing and factory upgrade program will let you scratch it. See **NEW PRODUCTS** for details.

And right in line with pushing the limits of performance, Charley Dobson gives us his "Z80 Slave Processor for Cromix." This product, from **Systems Atlanta** has caused a great deal of excitement, and we'll be following it up next issue with an in-house review.

To round it out, we've got a review of a very nice piece of work, the **T/H/C MAILLIST** and **T/H/C LABELS** programs, which appears in Pat Glenham's article, "Managing Your Mail Lists." There's also a new installment of Paul Hentzel's "Winning at Copyrights" which explains how one goes about correctly placing a copyright notice on your software. And then there's ... well, nevermind, you'll find out for yourself. Happy reading!

Bill Jaenicke
Editor

ONLY ONE COMMUNICATIONS PROGRAM HAS BEEN CALLED EASIER TO USE THAN CROSSTALK, SUPPORTS RFILE, SFILE, 3102 TERMINAL EMULATION, AND A FILE TRANSFER PROTOCOL THAT'S NAMED AFTER A FROG.



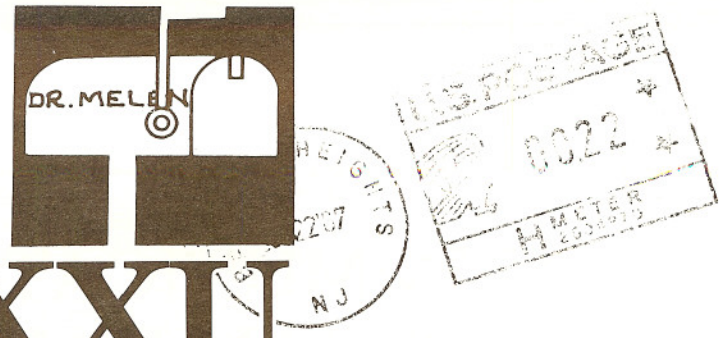
Yep, it's true. Only one program has what it takes to be set apart from the rest. Sure, we put in features like Xmodem, DEC VT-100 and VT-52 emulation, but we didn't stop there. If you're looking for a powerful communications program to help you share information with public information services, BBS systems, and your Cromemco systems, the answer is staring you right in the face.

Now all versions of ProCall, and ProCall/PC *plus* support the Kermit file transfer protocol, and we've made several other changes that won't go unnoticed either.

Ask your dealer about ProCall and ProCall/PC. Or ask us:

ProtoMatrix Software Development
1145 Park Heights Drive
Milpitas, CA 95035 USA (408) 263-8665

PSD



The XXU Letters

Editor's Note: *The following article was put together from sections taken from a series of letters between Dr. William Koch, a member of the Microcomputer Users Group of New Jersey, Delaware, & Eastern Pennsylvania and Cromemco's Dr. Roger Melen, Vice President, and Dr. Egon Zakrajsek, Senior Software Engineer. The correspondence took place between February and March of this year. It provides some additional interesting information in light of Dr. Stephen Huber's article, "Cromemco vs. the World."*

Dr. Koch wrote to Dr. Melen:

If Leon Uris hadn't thought of it first, this letter would have been titled "The Agony and the Ecstasy" of converting to Cromix-40. After many hours of collaboration with Antonio Ordonez and Chris Mukai to get the new hardware and software to work properly in a "loaded" system environment, I have finally been able to test Release 150 of Cromix-40 using four Octart boards as auxiliary Z80 processors. Biart boards were unreliable (consistently crashed after 2-3 minutes) or didn't work at all in my system configuration. System Atlanta's Z80 slave processor boards (internal Z80 program execution) were tested in comparison to the auxiliary Octart arrangement (Z80 processing external to terminal connection board and main CPU).

The ECSTASY is that the performance of the main XXU processor by itself is phenomenal. The AGONY is that some Z80 programs don't work at all, and of those that do, the performance of the auxiliary Octart/Biart is so poor as to make this "solution" for backward compatibility all but unworkable. Fortunately, the Systems Atlanta slave processor arrangement is a reliable and MUCH faster alternative which will allow us to continue using our older software in the new environment.

Here's a list of Z80 programs that DO work on auxiliary Octarts or Systems Atlanta Slave Processors:

Program	Vendor	Source	Aux Octarts	SA Slave
WriteMaster	Cromemco	Z80 Assembly	Slow (1)	Very Fast (6)
SpellMaster	Cromemco	Z80 Assembly	Slow (2)	Very Fast (6)
StatMaster	Cromemco	Z80 'C'	Slow (3)	Very Fast (6)
dBASE II	Ashton-Tate	?	Slow	Very Fast (6)
SuperCalc 2	Sorcim	?	Slower (4)	Very Fast (6)
ProCall	ProtoMatrix	Z80 Assembly	Slowest (5)	Doesn't run
StatMate +	Software Hill	PL/1	Slow (2)	Very Fast (6)

(1) Effective screen refresh is about 2400 baud. Character inserts are extremely tedious. Performance is marginal.

(2) Effective screen refresh is about 2400 baud = marginal.

(3) Characters entered from keyboard aren't echoed back until Carriage Return. Effective speed is about 2400 baud.

(4) Effective screen refresh is about 2400 baud = unworkable.

(5) Effective screen refresh is about 600 baud.

(6) Using the Systems Atlanta slave processor, programs are loaded into and run on the same board the terminal is connected to, saving the system CPU and memory for other chores.

Here's a list of Z80 programs that DON'T work on auxiliary Octarts or Systems Atlanta slave boards:

Program	Vendor	Language Source	Comment
RBTE	Cromemco	Z80 Assembly	Doesn't load
MCS Tape	MCS	Z80 Assembly	Bus Problem (7)

(7) Program menu is displayed, but execution stops when program tries to find the tape controller board. Controller board bus address (b0-b3) must conflict with something new to Cromix-40. HELP (!) Mark Byrd of Microcomputer Services is no longer supporting this program; any ideas?

I should emphasize that ALL of the above programs (except RBTE and MCS Tape) work properly and run FAST on Systems Atlanta slave processors under 11-, 20-, 30-, and 40-series Cromix. The Systems Atlanta approach is to connect terminals directly to the slave processor for internal I/O processing on an 8 MHz Z80H processor. Terminals run at a true 19200 baud speed (impressive compared to Octart terminal speeds even when the Octart is set for 19200 baud). Overall system performance is enhanced considerably since external I/O is held to a minimum. 68000 programs are passed on to the main system CPU for execution.

Another major advantage of using Systems Atlanta slave processors in a "loaded" environment is that there is almost no degradation in system performance when multiple users are running Z80 programs at the same time. For instance, it's not unusual for us to have 6 to 8 terminals running WriteMaster programs on slave processors, while the main system CPU is running 2 or 3 other jobs at the same time. Individual users are hardly aware that anyone else is using the system!

I can recommend the Systems Atlanta slave processor very highly. In good conscience I CANNOT recommend the external auxiliary Octart/Biart approach. If you have never tried a Systems Atlanta slave processor, I would strongly recommend that you purchase (or at least borrow) one for a test. Since completing our testing, we have upgraded our system to include 8 Systems Atlanta slave processors (Table 1). The overall system performance is truly phenomenal.

Here's a list of non-Z80 programs which DO work under Cromix-40 with or without auxiliary Z80 processors:

- Any Fortran program recompiled with 68020 Fortran (WOW!)
- CSCOPY (Cipher Systems: MS-DOS disk file transfer program)

And here's a list of other programs which DON'T work under CROMIX-40:

Program	Vendor	Language Source	Comment
Basic Plus	Cromemco	?	Need 68020 Basic
Kermit (8)	VSL Int'l	?	We need Kermit(!)
Ved (9)	Software Stds.	68000 C	Vendor notified

(8) Kermit is very important in our environment. This version, available free from VSL International (Switzerland), is somewhat incomplete (no server) but has been serviceable under Cromix 31.05.

(9) Ved is a communications package which incorporates VT-100 terminal emulation for the Cromemco 3102, C-05, and C-10 terminals. Very useful in a pinch. Software Standards should have a 40-series Cromix version ready very shortly.

Here's a list of bugs and suggestions for Cromix-40 improvements prior to the next SUDS release:

- Add /dev/z80/zio5 thru /dev/z80/zio8 device drivers.
- Link ed.bin and edit.bin to ce.bin, NOT screen.bin, in the /bin directory (also change links in help directory).
- Link spool.bin to print.bin and pr.bin in the /bin directory. To the uninitiated, this makes Cromix operate more like VAX and IBM mainframe computers.
- Bug: Forking-a-shell process from ce.bin locks up C-05 terminals. (No problem has been found using C-10 and 3102 terminals).
- A generic VT-100 profile is needed in termcaps. M100 and Z100 do not allow VT-100 terminals to use the ce editor properly. M100 comes close, but does not pass on cursor control. (Many scientists use VT125, VT220, VT240, or compatible terminals; we need more compatibility here).
- Provide a driver allowing the use of a Biart in place of PRI to run 3355B printers (frees up main CPU).
- A warning about a change in the interpretation of 64FDC switch settings is advisable. For those updating from 31.05 Cromix, they may not be able to use the boot disk until they move switch #5 to the UP position (all others DOWN).

Now for my WISH-LIST:

We are increasingly using Mac + computers as terminals into our VAX and IBM mainframe computers and would really like to use them as terminals into the Cromemco system. For all new programs using termcaps, we're almost there (the communication program we use, VersaTerm-Pro, makes the Mac think it is a REAL VT-100 terminal). However, there's no such luck for older Master-series programs requiring 3102 emulation (WriteMaster, SpellMaster, StatMaster). We would really like a 3102 emulator for the Mac (!!). (Then our management won't have to ask why we need two terminals per desk, and we won't have to explain why the Cromemco is still a critical part of our operation).

We would also like an AppleTalk connector so we can send data more easily between Cromemco and Macintosh computers and share the Apple LaserWriter printer.

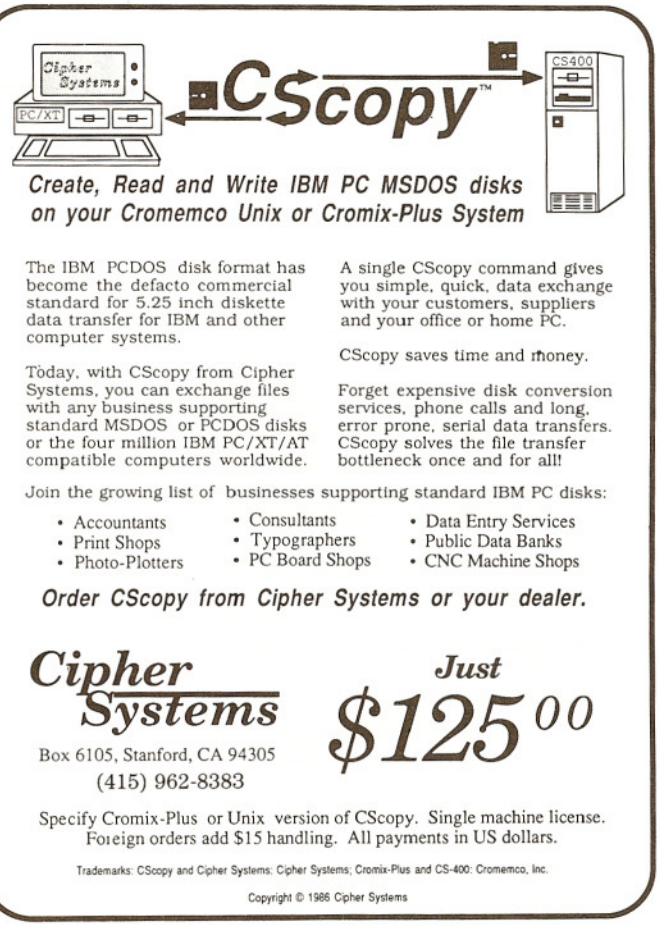
A Kermit for 40-series Cromix and a version of Basic compatible with Microsoft Basic would be really nice.

There are a large number of programs written for the IBM PC family of computers that use the Microsoft Basic interpreter. It should be a fairly easy task to bring these into the Cromix environment; I have already done this for two statistical programs using the Z80 (CPM) version of Microsoft Basic. Unfortunately Microsoft Basic is not the same as Structured Basic, Basic-Plus, or other versions of Basic that run in the 40-series Cromix environment.

Perhaps it would be possible for you to work with Microsoft to port their version of Basic into the Cromix environment. Alternatively, perhaps you could prepare a program that will translate Microsoft Basic code into Sbasic code. Here again you have the opportunity to open the Cromemco system up to a much larger audience by making programs available that run under a very common version of Basic.

Finally, we really need a 68000 (or 68020) version of WriteMaster and SpellMaster. The beauty of these programs is that I can train ANYONE to use them in 15 minutes or less. The 3102/CKBC style keyboards are extremely easy to use. And while relatively simple, these programs handle 99% of our current needs. Many of us who know and have used other programs (WordStar, Wang OIS, DecMate, All-in-One, Mass-11, Word Perfect, Microsoft Word) still prefer WriteMaster for the majority of our work. A newer version will allow us to maintain backward compatibility with our last 6 years of files.

As an alternative, I would settle for a 68020 Cromix/Unix version of Microsoft Word. It's already available in Unix and Microsoft is preparing it for Xenix. Of course. it's now the business standard in the PC and Macintosh world. How nice it would be to start work on a personal computer at home, then bring in the disk to finish editing with the bigger Cromemco system at work!!



FOR SALE

Cromemco System 3, Z-80, 64K RAM, 2 8" disk drives, custom desk mount. Intecolor 8001G 8-color CRT, 48x80 charac., graphics.

$3000 OBO.

San Diego 619-465-0839

TABLE 1: System Configuration

A. Systems Atlanta SAM2100 Power Supply (21-slot, HD power supply)
   1. 64FDC floppy disk controller
   2. 2048KZ memory board #1
   3. Octart terminal connection board
   4. (blank)
   5. Systems Atlanta Turbo Slave #1
   6. Systems Atlanta Turbo Slave #2
   7. Systems Atlanta Turbo Slave #3
   8. Systems Atlanta Turbo Slave #4
   9. Systems Atlanta Turbo Slave #5
   10. Systems Atlanta Turbo Slave #6
   11. Systems Atlanta Turbo Slave #7
   12. Systems Atlanta Turbo Slave #8
   13. Octart/Z-80 auxiliary processor #1
   14. Octart/Z-80 auxiliary processor #2
   15. Octart/Z-80 auxiliary processor #3
   16. PRI printer controller board
   17. STDC hard disk controller board
   18. 2048KZ memory board #2
   19. XXU main processor board (68020/68881)
   20. XMU memory management board
   21. (blank)

B. Disk Drives
   1. Tandon Dual 8" floppy disk drives
   2. Panasonic 5¼" floppy disk drive
   3. Cromemco HD50 hard disk drive (STD0)
   4. Cromemco HD150 hard disk drive (STD63)

C. Terminals

   8 (ea) Cromemco 3102 terminals —> Systems Atlanta slave boards

   3 (ea) Cromemco C-10 computers —> Systems Atlanta slave boards

   2 (ea) Cromemco C-05 terminals —> 64FDC, Octart boards

   1 (ea) modem —> Octart board

   3 (ea) network ports —> System Atlanta slave boards

   4 (ea) network ports —> Octart board

   ---

   21 active channels

   Network ports are shared by 4 (ea) 3102 terminals, 1 C-10 computer, and a host of DEC VT 241 terminals, IBM PC/AT computers, and Macintosh computers. Two additional outgoing ports allow communications/data transfer with mainframe VAX and IBM computers.

D. Printers
   1 (ea) Cromemco 3355A (NEC 7700 Spinwriter)
   1 (ea) Epson LQ1500 dot-matrix printer

E. Operating System Software

   40-series Cromix-Plus, release 150

--- *** ---

Dr. Egon Zakrajsek replied as follows:

This is a partial answer to the letter you wrote on February 13 to Dr. Roger Melen.

Biart Reliability:

Antonio set up a system with four Biarts. All Biarts smoothly ran Z80 programs though Antonio had to replace one Biart because that particular Biart did not work. Your problems must be attributed to occasional bad Biarts. Can you explain what you mean by a "Biart crash?'

Slow Z80 Programs:

Z80 programs do work slowly on the XXU. The reason is quite simple: every system function takes a lot of communications between the main and the slave processor (Biart/Octart). Programs that are quoted to be terribly slow are just doing things that are damn expensive. I believe that such programs can be changed a little to make them a lot faster, but I do realize that nobody is going to do it.

Z80 Programs Not Running:

All versions of Cromix, starting with 30.16 (40.16) up to release 151 (XPU or XXU) inclusive, have the same bug of which I was not aware until you pointed out that StatMaster does not work. The bug boils down to the fact that the indirect system call to execute the exit function acts as a do-nothing subroutine.

The Z80 C object time library does use this mechanism to terminate a program. As StatMaster is written in Z80 'C' (not in assembler) the StatMaster behaves as you described. Any other Z80 programs written in Z80 'C' have the same problem, though the manifestations might be quite different.

I am sending you the copy of Z80.bin (source 1.7). You should use it to replace the Z80.bin (source 1.6) in your /bin directory. I would appreciate it if you can try the Z80 programs that do not work again and let me know what happens.

68000 Programs That Do Not Work:

We do have an XXU version of Sbasic. Kermit is on our wish-list. As far as other 68000 programs that do not work, there is probably nothing I can do.

Bugs and Suggestions:

We do not want to put too many ZIO devices into the /dev/z80 directory. You are probably the only user who has more than four. Additional devices can be added by the system administrator at any moment.

The system administrator can also make any links or he can rename any files in the /bin directory if he finds it useful.

The problem of forking a Shell from the CE editor boiled down to a bug in the C-05 firmware. Most likely we will not be able to release a new version of C-05 firmware, so here are two suggestions to handle the problem:

After the C-05 is once reset from the keyboard, the problem disappears. At any time after power ON enter: CONTROL-SHIFT ? (question mark).

Disable the C-05 specific screen select feature in the termcaps file definition for the C-05 terminal (sq, sw, and sd).

The termcaps file we distribute with Cromix Plus contains the descriptions of the terminals we actually have in R&D.

We're Interested!

We're Really Interested...in what you have to say. Especially about how you use your system...the problems encountered and the solutions effected...unusual uses or environments...and any practical applications you would be willing to share with fellow members. These can be short notes for departments like 'bits & bytes...' and 'Tec Tips,' or full feature articles.

Contact Bill Jaenicke at I/O News for editorial guidelines or assistance. We're interested in unleashing your literary talents.

There is no way for us to write the termcaps description for a terminal we do not have. Even if we had the documentation we would not dare to write it without trying it. Termcaps additions could presumably come from the customers. Again, if we put customer's descriptions into termcaps and it then turns out to be incorrect, it would be too messy to fix it.

On the other hand, if you have any reason to believe that the distributed termcaps do not work with the terminals as described, we will be glad to fix it provided you can point out what does not work.

A printer driver running in a Biart is just an expensive way of printing. We do not consider the amount of work to outweigh the benefits. (Printer drivers run almost for free).

Wish-List:

There is no room in WriteMaster to do anything. The only reasonable solution would be to rewrite it and that means a lot of work. The same is true for SpellMaster and StatMaster.

I do not know who could and/or should do the 3102 emulator for the Macintosh.

An AppleTalk connector will be physically available on the XFDC. Today, we have no drivers for it. If the pressure mounts we will write them.

--- *** ---

Dr. Melen wrote:

I have reviewed your letter of Feb. 13 with Egon Zakrajsek, a member of our technical staff. He has sent you an updated version of software to correct some of the problems you cited, such as StatMaster not working. I hope these fixes are useful to you.

I think that you accurately capture the anguish of the conversion of a "loaded" system to 68020 processing. In addition, you demonstrate that not only is it possible, but able to yield exciting results.

I would like to see your letter published, with the appropriate changes made to incorporate the most recent fixes for StatMaster, etc. I hope you are now able to run the Whetstone benchmark which is 160 times faster than on the original 4 MHz processor (which was thought fast at the time of introduction).

I suggest that you send the modified letter to Bill Jaenicke at I/O NEWS.

Thank you again for your patience during this process.

--- *** ---

Dr. Koch's reply to Dr. Zakrajsek's letter:

Thank you very much for your letter of March 16 and the revised copy of Z80.bin. I found that this overcame some of the major problems we had with StatMaster and Statmate/Plus, but StatMaster still has a problem where characters entered on the keyboard are not echoed back to the screen until you press a carriage return. I could not prevent this problem from occurring by selecting "mode im" prior to executing StatMaster.

I was delighted that Nic Ivancic recently sent me his newest revision of the CE editor and a revised termcaps file. I found that this works flawlessly on our VT240 terminals in either 80 column or 132 column mode by using the instruction "TERM V132." This change in the CE editor and in the termcaps file now makes our VT100 and VT240 terminals far more compatible with the Cromemco system.

I am excited about your note that an AppleTalk connector will be available on the XFDC board. I mentioned this at the New Jersey/Pennsylvania/Delaware Micro Computer Users Group Meeting last week. You will be surprised how many people have interfaced Macintosh computers with their Cromemco systems. I hope you plan to provide drivers for this soon.

FAMILY HISTORY & GENEALOGY

Genealogy programs to help organise your family tree:

ROOTS/M by Commsoft for CDOS **$ 69.95**

ROOTS II by Commsoft for MS-DOS **$225.00**

Best of CDOS Public Domain S/W
Vol. 8: dBase II Genealogy programs **$ 25.00**

All prices are in U.S. dollars and include air mail delivery. Manuals are included with the ROOTS software. Extensive documentation can be found on the Volume 8 Disk.

Payment to be by check or bank draft in US dollars drawn on a United States bank, or in Australian dollars drawn on an Australian bank.

Send orders with payment to:

Applied Environmetrics
118 Gordon St.
Balwyn, Vic. 3103
Australia

FOR SALE
CROMEMCO SYSTEM III and SYSTEM ZERO

with 2 recently overhauled PerSci 299B dual drives and a spare 299B. Also, a brand new WYSE 350 color CRT and a Malibu Dual Mode 200 NLQ printer.

Boards include:

QTY	NAME	DESCRIPTION
1	DPU	Z80/68000 dual processor board
2	256KZ	256K RAM memory boards
2	64KZ	64K RAM memory boards
2	64K	RAM memory boards (Measurement System)—configured for Z80 Cromix
1	Seagate	ST-225 25-Meg. hard disk
1	STDC	hard disk controller (recently overhauled)
2	TU-ARTS	Serial/parallel communications board
2	16FDC	Floppy disk boards (1 recently overhauled)
1	4FDC	Floppy disk board
1	PRI	New style printer interface
1	PRI	Old style printer interface
1	ZPU	Z80 processor board
3	Misc	hard disk controller boards
1	Dazzler	
1	Joy Stick	Controller
1	SCC	Single Card Computer (Z80)

Cromemco Software:

CROMIX 68000 operating system
CROMIX 68000 'C'
CROMIX Z80 operating system
CROMIX Z80 'C'
Z80 Ratfor with Fortran IV
Z80 32K Structured Basic
Z80 COBOL
Z80 Word Processing System
Z80 Trace System Simulator
Z80 Macro Assembler
CDOS Z80 operating system

Miscellaneous software from other sources. All instruction manuals go with purchase. To the best of my knowledge, everything works except the spare 299B. Take the lot for $5000 or MAKE AN OFFER.

Write or Call: **Bill Haygood**
P.O. Box 577
Marina, CA 93933-0577
(408) 883-2743

Managing Your Mail Lists

by Pat Glenham

A third-party mail list software package is available for both Cromemco C-10 and Cromix systems, as well as MS-DOS systems, from The Hutchison Company based in Anchorage, Alaska. The **T/H/C MAILLIST™** and **T/H/C LABELS™** package, version 3.04 and 6.09 respectively, is a user-friendly set of programs designed to minimize the effort required to produce customized form letters and labels and to maintain mailing databases. And as stand-alone programs, they do not require any additional language to run.

The MAILLIST data entry program has features which enhance and ease the use of Micropro's Mailmerge™ program; the LABELS program was written to work with MAILLIST and Micropro's Datastar™.

I have been using the T/H/C package for almost two years and have found it to be extremely flexible for the various types of correspondence and mail lists I use in my publishing business, including my extensive distribution lists and 10,000-record subscription database.

MAILLIST is primarily a data entry program. The data entry process is made easier by defining the data items in an entry format file, then prompting for them individually, using a default entry format which can be changed to match your most frequent use.

Defining or changing entry formats is an easy process: MAILLIST creates Mailmerge format files and a master print control file which, in combination, virtually eliminates the need for any "dot" commands in your document files. This makes it possible for a new or occasional user to be productive right away. The only exception is if the document contains tabular data; even then, just two simple commands (one at the beginning of the table and one at the end) suffice.

The master print control file prompts you for the names of three files: the data file, the Mailmerge format file (created by MAILLIST), and the document file. To print form letters, you simply select M (Mailmerge) from the Wordstar no-file menu, type MMMaster as the name of the file to print, and answer the questions. For maximum flexibility, MAILLIST allows the user to vary the form letter as needed—from character width and page length to left and right margin justification and page offset, etc.

MAILLIST also provides a Tutor option which displays information and suggestions on the use of the program. MAILLIST is not, however, a word processing program. If changes must be made to a data file, Wordstar or some other word processing program must be used. The files which are created or changed by MAILLIST are entry format files, Mailmerge format files, data files, and the Mailmerge master print control file. These are all text files which can be printed or edited by the use of a word processing or text editing program.

MAILLIST provides an option to check a data file after changes have been made. You can verify that all records have the proper number of data fields without having to print a report to look for a misplaced quote or comma.

In addition to use with Mailmerge, MAILLIST is designed to be used with T/H/C's LABELS program to create mailing labels from MAILLIST databases. The database created with MAILLIST for Mailmerge is the same as the one used to create your mailing labels, thus eliminating needless duplication of records.

LABELS provides for printing labels up to 12 across from either a data file or text which you can enter or a combination of the two. The ability to serially number the labels is also included. Additional features include the ability to bold print a label line, center a line on the label, and to leave in or take out blank lines. Some examples of uses for labels defined from text only would be to create in-house postcards, cable numbering, return addresses, pricing, property stickers and so on.

For postcards, for example, you can define up to 21 printing lines on regulation postcards to communicate with your cur-

Continued

Dildine Industries can meet ALL your computer needs.

Dildine Industries is in it's ninth year as a Cromemco Dealer.

I have seen many computer companies come and go. I have seen many dealers come and go—some of them Cromemco dealers. I, on the other hand, am staying. I am committed to selling, servicing, and supporting Cromemco equipment. You can count on me to be around when you have a problem you need help with.

Dildine Industries is Hardware Know-How

I have hands-on experience with many computers ranging from the Altair 8800 to the Cromemco 420. Practical, working experience on PCs to mainframes. I have the technical knowledge to help you quickly solve your toughest hardware problems.

Call 800-262-0203

If you have had trouble finding a computer professional who can provide you with the answers you need and the support you want, call me, Kim Dildine, president of Dildine Industries. I believe you'll find my knowledge and expertise with Cromemco equipment a valuable resource. Straight talk and practical, working solutions to your computer problems.

Dildine Industries, Inc.

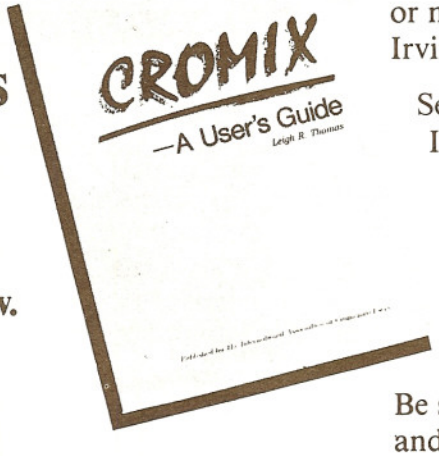
P.O. Box 4189
Hammond, IN 46324
In Indiana and from overseas call:
219-931-0203
Formerly K.I.D. Enterprises, Inc.
Cromemco dealer since 1978

Attention CROMIX Users

The book you've been expecting is finally here.

CROMIX — A User's Guide can be ordered now.

(While They Last)



Leigh Thomas' excellent tutorial on CROMIX, reprinted in the U.S., can now be ordered directly from I/O News.

The price is $25.00 plus $4.95 Shipping & Handling per copy. For bulk orders of 10 or more, the price is $17.75 per copy F.O.B. Irvine, California. California residents add 6% sales tax

Send Check, Money Order, or Credit Card Information to:

The I.A.C.U. - I/O NEWS
24843 Del Prado, Suite 473
Dana Point, CA 92629-2852

Be sure to include Membership Number and full shipping instructions with order. (Amounts noted are in U.S. dollars drawn on U.S. banks, only.)

This is a MUST BOOK for any CROMIX user.

MAILLIST

Continued

rent customers, to make mention of new products, or to reach potential customers. In other words, this T/H/C program is well suited to promoting itself!

For my lists, one option that has proved invaluable is the ability to choose a range within a data file to be printed. For instance, I can choose that only labels with addresses containing 'Seattle' will be printed or only those records within a certain range. This function also works in the report function, and as this is being written, Hutchison is updating the program to work an "everything not including X" range ability as well.

To run MAILLIST and LABELS, you need a display terminal with the ability to position the cursor on the screen and to erase from the current cursor position to both the end-of-line and the end-of-screen and a printer with a minimum width of 80 characters to print entry formats in the MAILLIST program. If your printer has a continuous forms tractor, it is much easier to achieve consistency when printing your labels.

Names of files accepted by the MAILLIST and LABELS programs are made up of the disk drive specifier (optional) and from one to eight letters and/or numerals. Data files may have a file type or extension appended to the file name. Automatic extensions on file names include .MLF for entry format files, .MMF for Mailmerge format files, .LFM for the labels format.

Operation of MAILLIST is controlled from a selection menu, with corresponding numbers and keywords followed by a brief description. The menu allows the user to: ADD records to a data file using the current entry format; CHECK that all records have the same number of data fields; DISPLAY or print the current entry format; EFORMAT to create or change an entry format; MASTER to create the Mailmerge master print control file; MFORMAT to create the Mailmerge format file; REPORT by printing or displaying data file records; SELECT a different entry format; SORT to reorder the data file; and TUTOR to receive a description of options and tips. Return exits from the program; ESCape will exit from any of the selections.

One nice option allowed by MAILLIST is the ability to enter default values at the prompt for any field which has a maximum length of more than one character by entering the letter D followed by an equal sign, then pressing the Return key. MAILLIST will then ask for the default value. The user types the default value just as if it were a direct entry for that field. When the program asks for the field data, the user types the letter D (upper or lower case) followed by a carriage return, saving valuable time. Another feature is the ability to repeat the previous contents of a field by typing the letter R followed by a carriage return. This has no effect on the default value.

The LABELS program is also menu-driven: FORMAT sets the printing format; PRINT accomplishes printing of the labels; READ loads a specific format from disk; SAVE will save your format for quick-reloading; and REVIEW allows you to see the current format.

Essentially, everything except the program is a straight text file and can be edited with either a screen editor, Wordstar or even Writemaster (if written out as a screen file). In some cases, this can be a faster edit—if the user remains aware of the correct number of fields for the chosen format. I usually use my screen editor when entering large blocks of corrections within the same file, however I find it's faster to use the menu-driven input format when there is a large amount of repeat information, such as "sales manager" as a generic salutation.

Because MAILLIST data files are simply text files in which records are delimited by a Carriage Return (or CR/LF) and fields within records are delimited by an ascii character (comma is default), database information can be "imported" from other database management systems, e.g. dBASE II™ and Informix™, or in-house designed applications. For example, Informix has an "unload ascii" option where a field delimiter can be specified. With this option one can use the powerful query capabilities of Informix to select records from a large Informix database and write these records to a text file in a form compatible with MAILLIST and LABELS.

I recommend that purchasers of T/H/C MAILLIST and LABELS first completely read the manual, which is fairly well written although not quite as user-friendly as the programs. However, The Hutchison Company has been updating the MAILLIST/LABELS program regularly, and the new manual-in-progress promises to be less technically worded. One fine touch in their manual is an excellent "error messages" section. And the index is well-documented and cross referenced.

Better than the best written manual, the technical support offered by Hutchison is superb; they have the PR sense to return phone calls promptly, as well as the telecommunications ability to do online troubleshooting when warranted.

The MAILLIST and LABELS package is available from:

The Hutchison Company
200 W. 34th Ave., #799
Anchorage, AK 99503

About the author: *Pat Glenham is a former technical writer who is now co-founder of Horizons Publishing, publisher of* PASSAGES MAGAZINE.

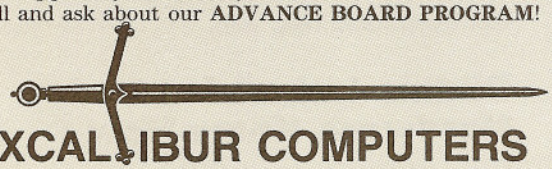
UPGRADE YOUR SYSTEM!!

256KZ to 1024KZ	**Qty 1**	**$ 650**
	2-4	**595**
	5-10	**475**
	11+	**400**
512MSU to 2048MSU	**Qty 1**	**$1,695**
	2-4	**1,270**
	5-10	**1,020**
	11+	**900**
For Sale:		
Excalibur Utilities	**Qty 1+**	**$ 99**
1024KZs*	**1**	**$1,200**
2048MSUs+	**1**	**$1,800**

90 day warranty on all modified boards.

* These are upgraded 256KZs
\+ These are upgraded 512MSUs

Need to upgrade your memory but can't afford the downtime? Call and ask about our **ADVANCE BOARD PROGRAM**!



EXCALIBUR COMPUTERS
4548 Auburn Blvd., Suite 191
Sacramento, California 95841
(916)971-9610

FOR SALE
CROMEMCO SYSTEM THREE (CS3A)
W/DPU, 4-256KZ, IOP AND 2 QUAD, SMDI,
64FDC, LARK II 50 Meg Removable, 2-C5, & 2-C10
Asking $8,500 or Best Offer
AUBURN BRASS INC./Santa Ana
Call Drew at 714-547-0333

1987 IACU General Survey

Feel free to be candid. The results of this survey will be used to improve the IACU and I/O NEWS. Many of the results will be shared with top management at Cromemco. This is your chance to speak out and be heard.

1 Operating Systems
- ☐ 01 C-10 CDOS
- ☐ 02 C-10 CP/M
- ☐ 03 PC-DOS
- ☐ 04 S-100 CDOS
- ☐ 05 S-100 CP/M
- ☐ 06 Z80 CROMIX
- ☐ 07 68000 CROMIX-D
- ☐ 08 CROMIX-PLUS
- ☐ 09 UNIX System V
- ☐ 10 Others: _____

2 Computer Models
- ☐ 01 C-10 PC
- ☐ 02 IBM-PC
- ☐ 03 PC-clone
- ☐ 04 CS-1
- ☐ 05 CS-Z2D
- ☐ 06 CS-Z2H
- ☐ 07 CS-3
- ☐ 08 CS-100
- ☐ 09 CS-200
- ☐ 10 CS-300
- ☐ 11 CS-400
- ☐ 12 CS-420
- ☐ 13 Others: _____

3 I/O and Communications
- ☐ 01 Biart
- ☐ 02 CNI
- ☐ 03 CSP
- ☐ 04 Dazzler Graphics
- ☐ 05 ENET
- ☐ 06 IOP/Quadart
- ☐ 07 IOP-X
- ☐ 08 Octart
- ☐ 09 PRI
- ☐ 10 SDI Graphics
- ☐ 11 S-Series Graphics
- ☐ 12 Tuart
- ☐ 13 Others: _____

4 Printers
- ☐ 01 Brother
- ☐ 02 Centronics
- ☐ 03 Cromemco
- ☐ 04 Diablo
- ☐ 05 Dataproducts
- ☐ 06 Epson
- ☐ 07 Facit
- ☐ 08 IBM
- ☐ 09 NEC
- ☐ 10 Okidata
- ☐ 11 Qume
- ☐ 12 Texas Instruments
- ☐ 13 Others: _____

5 Storage Controllers
- ☐ 01 4FDC
- ☐ 02 16FDC
- ☐ 03 64FDC
- ☐ 04 64FDX
- ☐ 05 WDI
- ☐ 06 WDI-II
- ☐ 07 STDC
- ☐ 08 STDC-X
- ☐ 09 SMDI
- ☐ 10 SMDX
- ☐ 11 Others: _____

6 Terminals
- ☐ 01 Ampex
- ☐ 02 Beehive
- ☐ 03 Cromemco 3102
- ☐ 04 Cromemco C-05
- ☐ 05 Cromemco C-15
- ☐ 06 DEC
- ☐ 07 Hazeltine
- ☐ 08 RGB
- ☐ 09 Televideo
- ☐ 10 Wyse
- ☐ 11 Others: _____

7 Memory
- ☐ 01 16KZ
- ☐ 02 64KZ
- ☐ 03 256KZ
- ☐ 04 512MSU
- ☐ 05 MCU
- ☐ 06 1024KZ
- ☐ 07 2048KZ
- ☐ 08 2048MSU
- ☐ 09 MCUX
- ☐ 10 32KBS
- ☐ 11 Others: _____

8 Processors
- ☐ 01 SCC
- ☐ 02 ZPU
- ☐ 03 DPU
- ☐ 04 Maximizer
- ☐ 05 XPU
- ☐ 06 XMM
- ☐ 07 XXU
- ☐ 08 XMU
- ☐ 09 Others: _____

9 Instrumentation & Control
- ☐ 01 4PIO
- ☐ 02 8PIO
- ☐ 03 ADC12
- ☐ 04 D+7A
- ☐ 05 DAC12
- ☐ 06 GPIB
- ☐ 07 Others: _____

10 Mass Storage
- ☐ 01 Floppy
- ☐ 02 Floppy
- ☐ 03 Hard Disk
- ☐ 04 Hard Disk
- ☐ 05 CTD Cartridge Tape *
- ☐ 06 9-Track Tape
- ☐ 07 Others: _____

11 Your Profession
- ☐ 01 Accounting
- ☐ 02 Agriculture
- ☐ 03 Broadcasting
- ☐ 04 Church Administration
- ☐ 05 Computer Sales/Service
- ☐ 06 Computer Sciences
- ☐ 07 Construction
- ☐ 08 Dentistry
- ☐ 09 Earth Sciences
- ☐ 10 Educational Institution
- ☐ 11 Engineering (Biomedical)
- ☐ 12 Engineering (Civic)
- ☐ 13 Engineering (Electrical)
- ☐ 14 Engineering (Mechanical)
- ☐ 15 Engineering (Structural)
- ☐ 16 Facilities Management
- ☐ 17 Financial Management
- ☐ 18 Government
- ☐ 19 Health Care
- ☐ 20 Hospital Administration
- ☐ 21 Insurance
- ☐ 22 Marketing
- ☐ 23 Medical Doctor
- ☐ 24 Medical Research
- ☐ 25 Military Services
- ☐ 26 Nursing
- ☐ 27 Pharmacology
- ☐ 28 Property Management
- ☐ 29 Publishing
- ☐ 30 Real Estate
- ☐ 31 Retail Business
- ☐ 32 Scientific Research
- ☐ 33 Software Development
- ☐ 34 Travel
- ☐ 35 Wholesaler/Distributor
- ☐ 36 Other _____

12 Software *(Indicate software you currently use under U and software you would like to know more about under W)*

U	W		U	W		U	W		U	W	
☐	☐	01 Accounting	☐	☐	14 Engineering	☐	☐	27 Library Science	☐	☐	40 Resource Management
☐	☐	02 Agricultural	☐	☐	15 Facilities Management	☐	☐	28 Management	☐	☐	41 Sales/Marketing
☐	☐	03 App. Development	☐	☐	16 Financial Management	☐	☐	29 Manufacturing	☐	☐	42 Service Industries
☐	☐	04 Arts	☐	☐	17 General Business	☐	☐	30 Media	☐	☐	43 Spreadsheets
☐	☐	05 CAD/CAM	☐	☐	18 Government	☐	☐	31 Networking	☐	☐	44 Statistics
☐	☐	06 Communications	☐	☐	19 Graphics	☐	☐	32 Office Automation	☐	☐	45 Tax Preparation
☐	☐	07 Construction	☐	☐	20 Health Care	☐	☐	33 Personal Computing	☐	☐	46 Text Processing
☐	☐	08 Data Conversion	☐	☐	21 Info Management	☐	☐	34 Process Control	☐	☐	47 Trade
☐	☐	09 Database Management	☐	☐	22 Insurance	☐	☐	35 Project Management	☐	☐	48 Transportation
☐	☐	10 Defense Electronics	☐	☐	23 Inventory Control	☐	☐	36 Publishing	☐	☐	49 Utilities (Gas, Power, etc)
☐	☐	11 Desktop Publishing	☐	☐	24 Lab Automation	☐	☐	37 Real Estate	☐	☐	50 Utilities (software)
☐	☐	12 Distribution	☐	☐	25 Language Processors	☐	☐	38 Recreational	☐	☐	51 Other _____
☐	☐	13 Educational	☐	☐	26 Law	☐	☐	39 Research			

13 IACU Feedback

01 How did you first learn of Cromemco?

02 How did you first learn of the IACU?

03 For what do you use your system(s)?

04 Besides yourself, how many people read your copy of I/O NEWS? _____

05 Indicate departments regularly read in I/O NEWS:
- ☐ INPUT
- ☐ OUTPUT
- ☐ BITS & BYTES
- ☐ NEW PRODUCTS
- ☐ C-10 ENCOUNTERS
- ☐ 32K CLASSROOM
- ☐ INSIDE CROMIX
- ☐ TEC TIPS
- ☐ SOFT TIPS
- ☐ SOFT TOOLS
- ☐ HACKER'S HOME
- ☐ USER'S NOTES

06 Which features or articles have you found to be most interesting/beneficial?

07 As above, but least interesting or beneficial?

08 What types of articles would you be most interested in reading? _____

09 As above, but least interested? _____

10 In general, do you find I/O NEWS editorial:
☐ Very useful ☐ Interesting only ☐ Not useful

11 What additional departments would you suggest (such as SOFT TOOLS)? _____

12 Would you be interested in contributing an article to I/O NEWS? ☐ Detail subject matter: _____

13 Have you responded to I/O NEWS advertising?

14 Which ones? _____

15 What were your general impressions? _____

16 Are there any specific side benefits you would like to see the IACU offer? _____

17 What new products would you like to see Cromemco introduce?
Hardware: _____
Software: _____

18 Do you have any suggestions on how the IACU and/or I/O NEWS could be improved? ☐ List: _____

Continued on back of page ...

ION-61

Continued

19 On a scale from 1 (low) to 5 (high), how would you rate Cromemco in the following categories?

	1	2	3	4	5
Overall Satisfaction	☐	☐	☐	☐	☐
Dealer Support	☐	☐	☐	☐	☐
Cromemco Service	☐	☐	☐	☐	☐
Affordability	☐	☐	☐	☐	☐
Performance	☐	☐	☐	☐	☐

20 Have you done any special peripheral interfacing (plotters, sensors, process control, etc)?
Describe: _____

21 How long have you been using Cromemco equipment? _____

22 Have you ever upgraded your system? ☐
Describe: _____

23 Are you considering an upgrade to your present system? ☐ Describe: _____

24 Further comments: _____

Mail this survey to:

The IACU - I/O NEWS
24843 Del Prado, Suite 473
Dana Point, CA 92629-2852
USA

Name: _____
Company: _____
Address: _____

IACU #: _____ **Phone:** _____

Attach Business Card

System Improvement Request Submission Form

Page 1 of _____

Submittor: **Firm:**

Address: **Phone:**

How to write a SIR:
Describe the capability you would like to see available on Cromemco systems. Be as specific as possible. Please don't assume that everyone knows how it's done on the XYZ system. Justify why the capability would be useful and give an example of its use. If you wish, suggest a possible implementation of your request.

Abstract:

Description and examples (use additional pages if required)

NEW PRODUCTS...

NEW PRODUCTS is a regularly appearing column devoted to announcing and following hardware and software products of interest to Cromemco users. Most information is derived from press releases submitted by vendors. As a result, I/O NEWS cannot be responsible for errors of ommission or any other inaccuracies.

Cromix-Plus Spreadsheet

Warecraft announces the first 68000 Cromix spreadsheet, **68Kalc**™. At last, Cromix users can get a full-sized, full-powered spreadsheet. Unlike competing programs, 68Kalc runs directly on the Motorola 68000 processor in Cromemco's Cromix-based computers. And, since it relies on Cromix termcaps for screen handling, 68Kalc will run on almost any terminal. Its commands mimic Lotus 1-2-3's, and online help is provided, so learning to use it is easy.

68Kalc is offered for $495.00, U.S. The price is refundable within thirty days to guarantee satisfaction. Direct inquiries to:

WARECRAFT
501 North 36th Street, #138
Seattle, Washington 98103 USA
(800) 227-3094

XXU UPGRADES

Since the introduction of its XXU card last autumn, Cromemco has received enthusiastic field reports on the performance of this processor. In some cases benchmarks have executed over 100 times faster on the XXU than on some of their earlier processors! Given the great success of the XXU and XXU-based systems, Cromemco is launching a new XXU sales offensive.

In particular, they want to make it easy for current customers to upgrade to the XXU. They will also be offering special summer pricing on XXU systems to make these new systems easy to buy.

New Upgrade Kits

Cromemco now offers two new upgrade kits (one for Cromix systems and one for UNIX systems) to upgrade older Cromemco XPU-based systems to the XXU.

The upgrade kit for Cromix systems is model XXU-UPGDC and includes:

- New XXU
- Cromix Plus-C operating system
- Upgrade parts and instructions for:
  - 64FDC-X
  - STDC-X
  - MCU-X

List price for XXU-UPGDC is just $4995, a savings of over $1500 from the price of the kit components purchased separately.

The upgrade kit for UNIX systems is model XXU-UPGDU and includes:

- New XXU
- New XMU
- UNIX-US operating system
- Cromix Plus-C operating system
- Upgrade parts and instructions for:
  - 64FDC-X
  - STDC-X
  - MCU-X

List price for the XXU-UPGDU upgrade kit is just $7495, a savings of over $3200 from the price of the kit components purchased separately.

Any Cromemco system produced during the last two years can be upgraded with these kits (excepting systems with ECC memory which must include an MCU card manufactured in the past year.) Systems manufactured in this time frame have cards with at least the minimum revision level required for upgrade. These minimum revision levels are:

Board	Minimum level required for upgrade
MCU-X	A
STDC	D, MOD 10
64FDC-X	E
OCTRT	B, MOD 5

The 1024MSU, 2048MSU, 1024KZ, and 2048KZ memory boards are all compatible with the XXU and do not require upgrading.

Systems with MCU, STDC, 64FDC, or OCTRT cards which are at earlier revision levels than those listed above will require the purchase of new replacement cards in addition to the upgrade kit.

New Factory Upgrade Program

To make it even easier for current users to upgrade to the XXU, Cromemco is also offering a factory upgrade service. Through this service any Cromemco computer meeting the revision levels described above will be upgraded at the factory for a special price of just $500 (effective until 12/31/87) over the cost of the upgrade kit alone. When a machine is returned to the factory for upgrade, the upgrade kit will be installed, the new operating system will be installed, the system will be fully tested, and the system will be returned with a new 90-day limited warranty. Contact your local dealer or Cromemco for details on this factory upgrade service.

To request this factory upgrade service, ask for service UPSERV-C for Cromix systems (price $5495) or UPSERV-U for UNIX systems (price $7995).

Special Summer Pricing for XXU-based Systems

To make it easier for customers to purchase new XXU systems, Cromemco is offering very special promotional pricing for selected models of their System 120 and System 420 during the summer months. These prices are effective from June 1 to August 31 and reflect enormous savings:

MODEL	LIST PRICE	PROMO PRICE	SAVINGS
CS120H50XXC40	16,996	13,995	3000
CS120H150XX40	20,995	15,995	5000
CS420H150XX40	29,995	25,995	4000
CS420H150XX40E	33,385	28,385	5000
CS420H300XX40	37,385	31,385	6000

IPC (Intelligent Process Controller) and New Octart Drivers and Enhancements

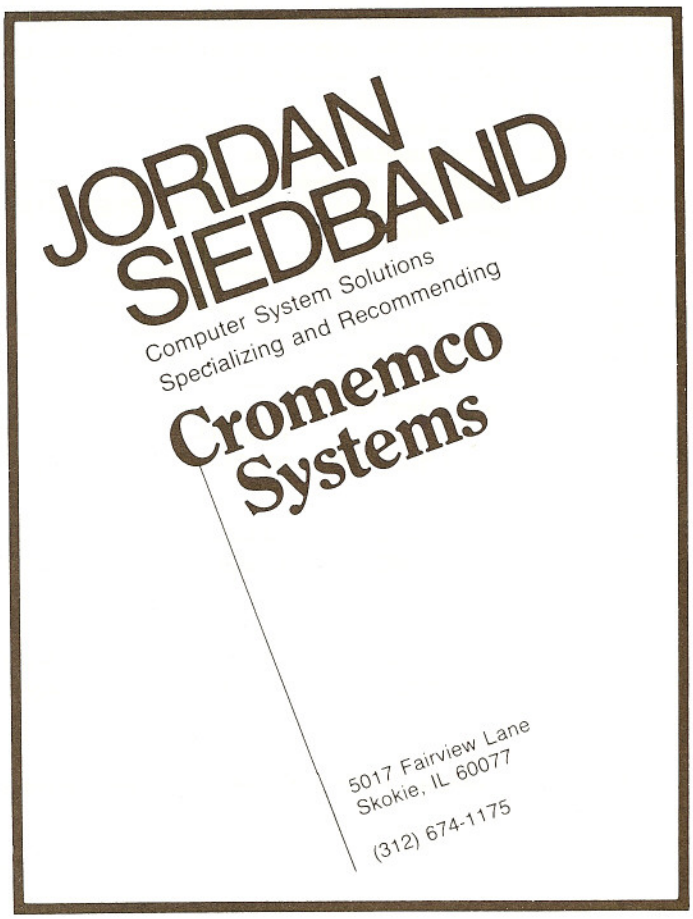
Microcomputer Specialists, Inc. of Grand Rapids, MI (formerly of Portage, IN) has been developing Hi-Tech I/O devices and drivers. The IPC supports up to 48K of high speed (70 ns) two-

port memory with a Z80H processor running at 8Mhz with zero wait states. The board also contains 4 serial channels with full hand-shaking (using optional personality boards) and 4 serial channels with just TX and RX. Also, on board is an 8255 which can support either two parallel printers (drivers optional and must be specified) or a Quick 2/SASI interface to which we intend to add a low cost 60 Megabyte streamer tape drive. For the past several years, this board has been used in distributed process control and data collection, front-ending many S100 UNIX/IDRS/CROMIX systems.

We have developed a driver to communicate with the IPC for Z80 Cromix which covers most of the basics for using it as terminal and printer support. When Cromix-Plus was released, we decided to support the IPC with its two-port memory in Cromix and Unix (Unix drivers are currently being completed). Once these drivers were completed, we decided to port them to the Octart, since so many users currently have them. During development, there were only two major items of concern; the slower clock rate of the Octart, and the fact that the Octart did not support two-port memory, which meant that all the data funneled through an 8-bit data port. We solved the first problem, by some degree, with a hardware speed-up modification, thereby increasing it up to 6 MHz.

Because the Octart does not support two-port memory, we first tested the IPC running at 4 MHz using a similar manner of communication. Even with the two deficiencies, the drivers had a major increase (approximately 300% at 4MHz) in throughput over the standard Octart drivers. The buffers in our drivers are 3K for output and 1K for input. All of these items are currently available as advertised. For further information, contact:

Microcomputer Specialists, Inc.
P.O. Box 88127
Grand Rapids, MI 49518
(616) 942-5412 or (616) 776-9646



UniLord™

Micro Consultants Corporation, Inc. (MCC), a Cromemco dealer based in Atlanta, Georgia (USA), has announced the release of a system administration utility for UNIX System V.2 called **UniLord™**.

Administration of a UNIX computer can require extensive knowledge about UNIX and how the system is organized. The purpose of UniLord™ is to provide the infrequent user with a windowed, menu guide through the steps required to perform day-to-day administration functions.

Installation of UniLord™ relieves the system administrator of knowing the intricacies of the individual commands without hindering anyone from performing system administration functions as before.

List price for the UniLord™ package is $595, and is available from:

Micro Consultants Corporation, Inc.
2687 Annex McCollum Parkway
P.O. Box 337
Kennesaw, Georgia 30144
(404) 423-0791

GRAFSMAN to Q-ONE Interface

SouthWind Software has announced that GRAFSMAN, a business graphics package for UNIX, now interfaces with Quadratron's Q-one word processor. This allows you to display both text and graphics on the same page on dot-matrix and laser printers.

Using Q-one's graphics reservation command to reserve the appropriate amount of space within the document and GRAFSMAN's word processing filter, text and graphics are merged on the printed page. On the screen, the user sees the box representing the amount of space to be used for the graph and enters the command which GRAFSMAN recognizes to place the graph in the text.

Any chart that you create with GRAFSMAN or the TACTICIAN spreadsheet can be included within any Q-one document. It is now possible to produce the kind of reports everybody has been wanting all along.

For more information contact:

SouthWind Software
4520 E. 47th St. So.
Wichita, Kansas 67210
(316) 524-9100

Laser Printer Support

TranScript* software source code, available from Adobe Systems, allows Cromemco UNIX users to access state-of-the-art laser printers. TranScript includes output filters and translators for troff files, plot files, Diablo 630 print files, Tektronix 4014 files, and simple text files.

Application software that produces these output files can now take advantage of PostScript-based laser printers with Cromemco Systems.

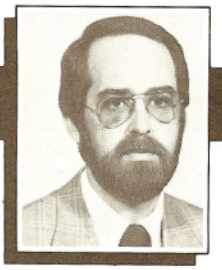
PostScript* software, also produced by Adobe Systems, resides in many popular printers, such as the Apple* LaserWriter*, and is a necessary companion for TranScript.

TranScript source code is available on 9- track tape for $1,795 from:

Adobe Systems, Inc.
1870 Embarcadero Road
Palo Alto, CA 94303
(415) 852-0271

*TranScript and PostScript are trademarks of Adobe Systems, Inc. Apple and LaserWriter are trademarks of Apple Computer.

Continued on page 39



SOFT TOOLS

SOFT TOOLS is a regularly appearing column dedicated to UNIX and Cromix users. Its aim is towards simplifying the administration and maintenance of multi-user systems. It is edited by Tom Ronayne, President of Advanced Programming Techniques Corp. (APTC), P.O. Box 19549, Detroit, MI 48219, (313) 835-0808.

Introduction

Cromemco provides us a Message of the Day (MOTD) facility in Cromix, and AT&T provides the same thing in UNIX System V. First thing we seem to do with the system after we've got it up, running, and doing work for people is shut the fool thing off because we get sick and tired of looking at the default message, or we stick something or other in the /etc/motd file whenever we get around to thinking about it.

I've always liked the little quotes you get on some calendars: you know, the cute little "lessons in life and living" that pocket planners, desk calendars, and wall calendars sometimes have printed on the "week" pages. I thought it would be kind of neat—which has absolutely nothing to do with being useful—to have a "saying" like the pocket calendars have pop up on the terminal when I log on. Oh, by the by, I also wanted it to happen automatically, say, every week.

Last time, I wrote about days and dates, with the subdtt, subttd, and subdow utilities for calculating the elapsed days from a date, calculating the date from the elapsed days, and calculating the day of the week from the elapsed days. I gave a couple examples of practical use of the routines, and, this time, I'm going to give a totally IMpractical use—it's just for fun.

motd Program

This motd program depends on the system-supplied date to do its thing. In Cromix-Plus, the Fortran-77 and C compilers give you direct access to the system date: in Fortran, the "fsyslib" routine getdat and in C, the "syslib" function getdate supply the month, day, and year in an integer array. Once you've got the month, day, and year it's no great trick to use a tool like subdtt to figure out what day of the week it is, what week of the year it is, and other things you need to know to make "something" happen.

Now, this motd program is trivial—all it does is change a "cute" message every week; but, the tool has other, practical, applications that may suggest themselves.

So, What's It Do?

I've got the motd program set up to change the message of the day once a week, and have decided that the week changes on Sunday. The givens are that there are 52 weeks in a year, the first week of the year starts on the Sunday less than or equal to January 1, and the "fifty-tooth" week is the last one no matter what day December 31 falls on.

FOR SALE

Cromemco System 3, 1024KZ & 256KZ (1¼ Meg Ram), XPU, STDC, IOP with 2 QUADARTS, 2-20 Meg Hard Disks, 2 Tandon 8" DSDD Drives, Cromemco 3102 Terminal, Cromemco 3355A Printer, Cromix-Plus, WriteMaster and Much More, Asking $4000 For All

(914) 592-5000

The data file, /etc/motd.dat, looks like this:

```
1       2
No one ever regarded the first of January
with indifference.
2       3
The rule on staying alive as a forecaster is to
give 'em a number or give 'em a date, but
never give 'em both at once.
                .
                .
                .
52      3
A painting in a museum probably hears more
foolish remarks than anything else in the
world.
```

The two numbers are the week number (1, 2, ..., 52) and number of lines of text in the message. The text is just plain text, but it's limited to a maximum of 45 characters (so it'll fit on about half an 80-column screen).

The file is set up this way because it's easier to sequentially read a file of this limited size with specific data in it; i.e., we'll identify what week we're looking for, and read until we find it. That way, you don't have to invent any sexy KSAM or ISAM tools (although, that's not a bad idea for a future effort).

This motd is set up for weekly changes, but, if you wanted to have a different message of the day every day, that wouldn't be hard, either—getting the messages in is your problem.

Listing 1 shows the code for the motd program that does the trick.

Listing 2 shows last time's subdtt, re-written as a function, dtt. We're using dtt (instead of subdtt) because the system call returns integers for the date—subdtt is a "human interface" routine that wants a string for the month name, and it's easier to build a new tool from an old one (in this instance) than it is to fiddle the month into a string form that we won't use anywhere else in the program.

System Date

I've (maybe) violated my own rules (see the sidebar) with this tool. I had to get the system date, and I had to use a library function to do so. Library functions—the ones that give you access to system functions from your high-level language—are extensions, and, as such, should be avoided like the plague. In this case, though, we don't have an "ANSI Standard" Fortran (or C, or Pascal, or Basic) intrinsic that gives us a date or time, so we can "kind of" get away with it.

How and Why It Works

In Cromix, the system date is returned in an integer array, dimensioned by four; here, we used an array, integer date (4). The lines,

```
if (getdat (date) > = 0)
   stop "Unable to read system date."
```

are the ones that actually call the system library routine to read the date from the operating system, and the one that is the "iron underpants" (if you can't read the date, you're in deep trouble). The date is returned in the array: date (1) is the day

of the week; date (2) is the year; date (3) is the month; date (4) is the day of the month.

We're trying to find out what week of the year this is, and we need to know two things to do so: what's the first day of this year, and what's today. We calculate the first day of this year:

```
first = dtt (1, 1, date (2))
```

and, because a week starts on Sunday, we adjust backwards to the previous Sunday:

```
while (mod (first, 7) >= 0)
  first = first-1
```

Then, we want to know what "today" is, so we can figure out what week this is:

```
nweek = dtt (date (4), date (3), date (2))
```

and

```
nweek = int ((float (nweek)-float (first))/ 7.0) + 1
```

do so.

Once we've got this information, we can just open up the data file, /etc/motd.dat, and read until we find the right week. (You may notice that you can use this "tool" to figure out pay periods, vacation schedules, and all sorts of other stuff.) Then, we write what we found into /etc/motd so the system will display it whenever somebody logs on.

UNIX System V System Date

In UNIX, everything works exactly the same as it does in Cromix-Plus, except for getting the system date. UNIX supplies the system date as the number of seconds since 00:00:00 January 1, 1970, and you've got to translate that into something you can use. In my own case, I use 1964 as a base date (I can't use 1970, because 1970 wasn't a leap year, and the Zeller's Congruence routines won't work), and I didn't want to re-write everything I've ever done that uses days and dates, so I wrote a "primitive" that emulates the Cromix-Plus getdat routine (and makes the program portable): the getdat function, **Listing 3**.

The getdat function needs a way to convert elapsed days to a date in the form expected by the motd program, and **Listing 4** (ttd) shows it.

Some may think that doing it this way is the hard way—why not just re-write the thing for UNIX? Well, I don't like re-doing things, and I abhor maintaining two (or more) sets of code that do the same thing. Too, I'm a Cromemco dealer, and I work in both the Cromix-Plus and UNIX System V operating systems (each has its own merits and demerits). I happen to think that the Cromix system interface routines make a lot more sense than the UNIX routines do—they're easier to use and give me great control with little effort; in a word, Cromix is more elegant. Something simple, like getdat, makes sense to write as a primitive. And, if a primitive can be written in a high-level—thus portable—language, so much the better.

How To Use It

In the first *Soft Tools*, we talked about a daily back up routine. Daily depends on the Cromix day function, and, basically, is a shell program that does "something" depending on what day of the week it is. In Cromix, you can have a little background routine running for automatically executing motd on the "right" day. **Listing 5** shows a suggested command file for changing the message on Sunday.

If you're using the previously-published "daily" routine, just stick the "/bin/motd" command line in the "Sunday" part of daily:

```
        .
        .
        .
%
%    none--no backup performed on selected days
%
%none
%
echo "No System Backup Today" > /dev/console
time > /dev/console
/bin/motd
goto wait
```

UNIX motd

UNIX, on the other hand, has a nifty routine called cron. Cron does things at times given in /usr/lib/crontab. You add a line in /usr/lib/crontab that executes the motd program on Sunday, every week. Much simpler.

We also stick a line to execute motd in the startup file: in Cromix, it goes in /etc/startup.cmd; in UNIX, it goes in /etc/rc.

motd.dat

The file that contains all the messages/etc/motd.dat — is not listed in the magazine; it's too long. But, don't despair — it's available, as are all the routines talked about here. You can contact *I/O NEWS* for a disk, or to down-load from the *I/O NEWS* Cromix system, or you can get all the files from us.

You may dial-in to our Cromix-Plus system at 300, 1200, or 2400 baud at (313) 835-0809 and transfer the files to your system (we suggest you use the free ccall utility supplied with all Cromix systems). Or, you can send us a blank disk (5-¼" or 8"), along with $5.00 (for postage and handling), and we'll copy everything for you. Be sure to tell us if you want CDOS, Cromix, UNIX, or MSDOS format, and number of sides and density (if you want to send a cartridge tape, that's fine, too—but be sure to tell us if it's an "ftcd" format or "ftp1" format).

FOR SALE

CROMEMCO CS-100 SYSTEM
WITH

8 Channel Serial Controller
C5 Terminal with CKBC
50 Meg Hard Disk
Stream Cartridge Tape Backup
IBM PC to Unix Communications
Informix Relational Database
Complete Documentation
$8,000 TOTAL

Mark Sutherland
404-449-6991

CROMEMCO SYSTEM III

Desk Model, four 8" Drives, ZPU, 64KZ, PRI & 16FDC. Includes 3102 Terminal and Keyboard, Mannesman Tally 1000 Serial Printer, Programs & Paper.

$2,500 OBO

Call Bruce 10 am—5 pm M-F

(714) 492-7324

Listing 1

```
define(STDOUT,1)
#       motd--display a new message of the day every so often
        program motd
#       variables
        character * 78 iname, oname     # input/output files
        character * 45 line1            # message line 1
        character * 45 line2            # message line 2
        character * 45 line3            # message line 3
        integer date (4)                # system date
        integer first                   # first day of this year
        integer lines                   # lines of text in message
        integer nweek                   # current week of the year
        integer week                    # this week
        integer ilun, olun              # files logical unit numbers
#       functions
        integer dtt                     # elapsed days conversion function
        integer getdat                  # system date function
#       initial data
        data ilun, olun /5, 6/          # logical unit numbers
        data iname /"/etc/motd.dat"/    # motd data file
        data oname /"/etc/motd"/        # motd file
#       program starts here
        if (getdat (date) ^= 0)         # get system date
                stop "Unable to read system date."
        first = dtt (1, 1, date (2))    # first day of this year
        while (mod (first, 7) ^= 0)     # back to previous sunday
                first = first - 1
        nweek = dtt (date (4), date (3), date (2))      # today
#       figure out what week of this year we're starting
        nweek = int ((float (nweek) - float (first)) / 7.0) + 1
        if (nweek < 1)                  # this should never happen
                nweek = 1
        else if (nweek > 52)            # neither should this
                nweek = 52
        write (STDOUT, *) "This is week number ", nweek, " of ", date (2) + 1900
#       open the data file
        open (ilun, file = iname, status = "old")
#       open the message-of-the-day file
        open (olun, file = oname, status = "new")
        for ( ; ; ) {                           # read messages till this one
                read (ilun, *) week, lines
                read (ilun, "(a)") line1
                if (lines >= 2) {
                        read (ilun, "(a)") line2
                        if (lines == 3)
                                read (ilun, "(a)") line3
                        }
                if (week == nweek) {            # this is it, change motd
                        write (STDOUT, "(/ a /)") "This is the new motd..."
                        write (STDOUT, "(a, t30, a)") "Thought for the Day:", line1
                        write (olun, "(a, t30, a)") "Thought for the Day:", line1
                        if (lines >= 2) {
                                write (STDOUT, "(t30, a)") line2
                                write (olun, "(t30, a)") line2
                                if (lines == 3) {
                                        write (STDOUT, "(t30, a)") line3
                                        write (olun, "(t30, a)") line3
                                        }
                                }
                        break
                        }
                }
#       close the files
        close (ilun)
        close (olun)
        end
```

Listing 2

```
define(BASE,64)
#       dtt--convert date to elapsed days
        integer function dtt (day, month, year)
#       input is a date in the form DDMMMYY
#       output is the number of days since march 1, 19-BASE year
#       specifications for calling variables
        integer day, month, year
#       specifications for internal variables
        integer daymth (12)
        integer iy, i, m
#       number of days in each month
        data daymth /31, 29, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31/
#       routine starts here--set internal year
        iy = year
#       assign month to m
        m = month
#       calling month and day of the month any good?
        if (m <= 0 | day - daymth (m) > 0 | day <= 0) {
                dtt = -1
                return
                }
#       account for leap years
        if (day - daymth (m) == 0)
                if (m - 2 == 0)
                        if (mod (iy, 4) ^= 0) {
                                dtt = -1
                                return
                                }
#       watch for century rollover
        if (iy <= BASE)
                iy = iy + 100
#       years since 19-BASE year
        iy = iy - BASE
#       months since march
        m = m - 3
#       get correct year and month
        if (m < 0) {
                m = m + 12
                iy = iy - 1
                }
#       calculate days since 01marBASE
        dtt = int (float (iy) * 365.25) + int (float (m) * 30.6 + 0.5) + day - 1
        return
        end
```

Listing 3

```
#       getdat--get unix system date (emulates cromix function)
        integer function getdat (buf)
        integer buf (4)                 # array to hand back to caller
        integer ndays                   # number of days
        integer ttd                     # conversion function
        real dsecs                      # number of seconds in a day
        real tloc                       # local time
#$include /usr/include/libfs.dcl         # library definitions
        data dsecs /86400.0/
        tloc = float (time (ndays))     # get seconds from 00:00:00 01jan70
        tloc = tloc - 18000.0           # offset to EST from GMT
#       calculate number of days since 01jan70 and add difference in
#       days between 01mar64 and 01jan70
        ndays = int (tloc / dsecs) + 2132
        buf (1) = mod (ndays, 7) + 1    # day of the week, same as cromix
#       convert number of days into a date
        getdat = ttd (ndays, buf (4), buf (3), buf (2))
        return
        end
```

Listing 4

```
define(BASE,64)
#       ttd--convert elapsed days since 01mar64 to date
        integer function ttd (ndays, day, month, year)
        integer n, ndays, day, month, year
        if (ndays < 0) {                        # number of days good?
                ttd = -1
                return
                }
        year = float (ndays + 1) / 365.25       # years since base year
        for ( ; ; ) {                           # days since march 1
                n = ndays + 1 - int (float (year) * 365.25)
                if (n > 0)
                        break
                year = year - 1
                }
        month = int (float (n) / 30.6)          # month this year
        for ( ; ; ) {                           # day of this month
                day = n - int (float (month) * 30.6 + 0.5)
                if (day > 0)
                        break
                month = month - 1
                }
        month = month + 3                       # correct the month
        if (month - 12 > 0) {
                month = month - 12
                year = year + 1
                }
        year = year + BASE                      # year offset
        if (month <= 0)
                month = 3
        if (year > 99)                          # watch for century
                year = year - 100
        ttd = 0                                 # everything ok
        return
        end
```

Listing 5

```
%       initialize /etc/.motd2 file for this shell routine startup
day > /etc/.motd2
%       wait--check for change in the day-of-the-week
%wait
%       put the day in file /etc/.day1 for comparison
day > /etc/.motd1
%       compare /etc/.motd1 and /etc/.motd2 for match, result in /cdm/.motd3
cmpasc /etc/.motd1 /etc/.motd2 >* /etc/.motd3
%       get the first line of /etc/.motd3
input < /etc/.motd3 > /etc/.motd1
%       if the first character is "/", there's no change of day
testinp -f /etc/.motd1 /
%       the day did change, transfer
if -err goto doit
%       update /etc/.motd2 with the current day-of-the-week
day > /etc/.motd2
%       put the process to sleep
sleep 9000
%       check for roll-over to new day-of-the-week
goto wait
%doit
testinp -r /etc/.motd2 "Today is Sunday."
if -err goto changit
goto wait
%changit
/bin/motd
goto wait
```

Rules of Tools

At the outset, I made a conscious decision to never present any tool in assembly language (even if it would be easier) for the simple reason that anything written in assembly is not portable to any machine other than one with the same processor as the original machine. I also decided that I'd never present anything in Structured Basic (no matter how "good" it is) for exactly the same reason. The tools we're using for this MOTD "toy" are all written in high-level language; in fact, they were originally written in RatFor (Rational Fortran) for use on a Z80 CDOS system, and they "ported" directly to 68000 Cromix—and to UNIX—when 68000 systems became available to me with *no changes whatsoever* because I'd written them to be as portable as possible. (I've since modified the routines to take advantage of the more elegant I/O format capabilities of Fortran-77, though.)

Portability—where you can compile source code on a different machine than the one it was written on and it "goes"—is a difficult goal at times. I've found Fortran to be more "portable" than C (and Pascal is a basket case), but the goal of portability may be achieved if you work at it. Portability also means that, when you upgrade your operating system from release to release, your original code still works without re-compilation (the "innocent bug" in the SVS products "found" by memory management in Cromix-Plus version 31.40 is the singular exception to portability I've ever found in a Cromemco software product). [Editor's Note: the bug alluded to is "fixed" with the patchbug utility that comes with Cromix-Plus versions 31.40 and 31.51].

Half the battle is having access to American National Standards Institute (ANSI) compilers. Those of you who have been around for a while will remember the difficulties everybody encountered with IBM's Fortrans: IBM has the distinctly bad habit of going its own way, shoving things down its customers' throats, changing those things, and then shoving the changes down customers' throats. (The Pretend Computer—which seems to change every week—is an excellent example; why anybody ever buys anything from IBM is beyond me.) The bald fact is that if a compiler doesn't say "ANSI Standard" on it, it ain't—and you're taking a big chance with all your efforts.

The other half of the battle is to **never** use the "extensions" compiler manufacturers seem determined to include in their products—remember, the goal is portability, which can't be achieved if you use, say, DEC's Fortran extensions; they don't work on other machines.

In the Fortran "world," we've got Fortran-66 and Fortran-77 as ANSI standards: you write something in Fortran-66 and it'll work on any other machines that support Fortran-66 (ditto with Fortran-77), and, anything written in Fortran-66 will "compile and go" in Fortran- 77. See, if it **says** Fortran-66 or Fortran-77, its got to *be* what ANSI said those things are. A product from, say, MicroSoft doesn't **say** Fortran-77—it says "MicroSoft Fortran," and, in the fine print, it says "a sub-set of the ANSI standard," and there is usually a large number of pages describing the "departures from the ANSI standard," and you, folks, are taking a big chance if you are trying to write portable code using one of these compilers. You gotta wear your iron underwear!

Pascal is a basket case. Flat out, there is no standard for Pascal. There is Borland International's **Turbo Pascal** (which is the "standard" on Pretend Computers), and there are other implementations with pretensions. The sad fact is, Pascal was never intended to actually be used on computers—it was invented as a way of teaching "structured code" to computer science students. My own study of this "language" leads me to believe that it's better left to blackboards.

In the C world, you've got a problem. Most everybody conforms with the "Kernighan and Ritchie" de-facto "standard" to some extent (the key phrase is "to some extent"), but there isn't (at this writing) an **ANSI** standard. There is a **proposed** ANSI standard, but one doesn't exist today. As close as we can get is the Silicon Valley Software (SVS) C compilers that Cromemco sells and the AT&T compiler included in the UNIX Programmers Tools. (For my money, the AT&T C is a better bet for portability—particularly between UNIX installations—but SVS C is pretty ok, too. AT&T has a better chance of influencing the ANSI committee—in the "right" direction—than most other C manufacturers. I've got both of them, use both of them, and—in **most** cases—can't tell the difference.)

Generally, I've found that if I write something in AT&T C it ports to Cromix, but that SVS's C doesn't include many of the "standard" utilities of the AT&T library. They're easy enough to write (they're all in "The C Programming Language"), but it's sort of a pain to not have things like "atof" included in the library.

Here's one other thing you can count on: if Enormous Azure gets involved, you can be sure that the results will be the lowest common denominator (think about it). If you care about what you use to make your living, it would be a real good thing for your future in computing if you'd get involved in the setting of standards and lobby like mad for AT&T's implementation of language: write to ANSI, find out what's going on, and, if you don't agree, say so.

The foregoing diatribe is to make a point: watch your hiney.

CD

Model 525CT Cartridge Tape Drive. Complete unit—drive, cabinet, power supply, and cables. Uses 3M DC-600A (or equivalent) cartridge tapes and works with all Cromix-Plus and UNIX System V cartridge tape utilities (inittape, ftar). Use with 64FDC only, $995 (plus shipping).

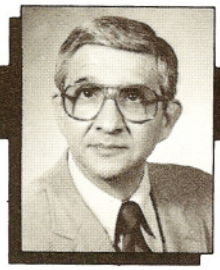
Model 85MST506 85M byte External Winchester Drive. Complete unit—drive, cabinet, power supply, cables, formatted with your choice of either Cromix or UNIX file system. 85M byte drive for use with STDC only, $1,995 (plus shipping).

Cartridge Tapes:

Quantity	Price Each
One	$ 29.50
Box of 5	27.50
Box of 10	25.50

Prepaid Only

ADVANCED PROGRAMMING TECHNIQUES CORPORATION
P.O. Box 19549 • Detroit, Michigan 48219
(313) 835-0808



32K CLASSROOM

32K CLASSROOM is a regular column aimed at explaining programming techniques using Cromemco Structured BASIC. The column is edited by Bernie Thomas. Users are invited to submit programming examples and editorial contributions to Bernie Thomas at P.O. Box 50119, Nashville, Tennessee 37205.

First of all, I'd like to say I regret not having an article in the last two issues of *I/O NEWS*, but the company, for whom I worked, relocated, and I chose not to make the move. This resulted in a career change for me, which resulted in a temporary shortage of about 10K of RAPT, and about an equal amount of RAC. For those who are not familiar with the terms, RAPT is Readily Available Power of Thought, and RAC is Readily Available Cash.

One of the career opportunities I have chosen to pursue is to design and write a software package for small to medium sized manufacturers. I picked the Cromemco System 100 using the Cromix-Plus Operating System and 68000 Structured Basic.

Among the most important goals of the system is to eliminate the need of a highly trained in-house person to support the system. The ability to execute a Shell Command from Basic is a very simple way to accomplish this.

If you are using 32K Basic, you can still take advantage of the features of the Shell Command by patching your version of Basic per the instructions of Norman Miller, which appeared in *I/O NEWS Volume II, Number 2*. Before I began using 68000 Basic, I used Norman's procedure in a similar fashion.

For example, backing up the data files to small floppy disks, using **Cptree** with the **-t** option, by directory, is normally done from the Cromix level by someone who is familiar with the commands. In my system, all data files are in directories by category: **/data/sales**, **/data/purchasing**, etc. First of all it is important that the correct disk is mounted when backing up a particular directory. I have accomplished this task using the **Sh** Command from Basic, and a Basic Routine which reads a label file on the disk to insure that the correct disk has been mounted.

To illustrate, let's suppose we wish to backup the **/data/sales** directory to the proper floppy disk. Assuming this is the first time this task has been performed, we must run the **Initflop** and **Makfs** Commands. In addition a Label File must be created on the floppy.

FOR SALE

Cromemco System 100 CS100H50X20 2 MB ram, OCTART, PRI, 50 Mb HD, a C10 terminal and TVI 950 terminal. Software includes latest version of UNIX and CROMIX PLUS. Other software includes UNIX C and CROMIX C compilers, Z80 Assembler and Basic. Unused/w registration: Informix Relational Database, UNIX COBOL Compiler and Forms-2, UNIX Documentor's and Programmer's tools. $6000.

317-452-6881 after 5 PM. EST


```
100   Dim Del40$(39) : Del40$=" "+Del40$(-1)
110   Dim Label$(9)
120   Dim File$(34)
130   Dim Problem$(2)
140   Dim Error$(10)
200   Gosub Crt'clear : Init'try=0 : Makfs'try=0
210   R=5 : C=1 : Gosub Loc : @"You May Init New Disk For"
220   @ Tab(25);"1) Purchasing Data Files"
230   @ Tab(25);"2) Engineering Data Files"
240   @ Tab(25);"3) Sales Data Files"
250   @ Tab(25);"4) Production Data Files"
260   @ Tab(25);"5) Accounting Data Files"
270   @ Tab(25);"6) EXIT"
280 *Ask : On Error Stop
290   R=12 : C=1 : Gosub Loc
295   Input"Enter the NUMBER of your selection >",I$
300   Gosub Loc : @ Del40$;
310   On Error Goto Ask : I=Valc(I$) : On Error Stop
320   If I<1 Or I>6 Then Goto Ask
330   Gosub Crt'clear
340   If I=6 Then Run"menu1.sys"
350   Data"Purchasing","Engineering","Sales","Production","Accounting"
360   Restore 350 : U=0 :   Repeat : U=U+1 : Read Label$ : Until U=I
370   R=10 : C=1 : Gosub Loc
380   @"Place a New Disk, Labeled ";Label$;", in the Disk Drive"
390   @ : @"Holding the Disk in your RIGHT HAND with";
395   @"the Label under your THUMB,"
400   @ : @"and the OBLONG SLOT AWAY FROM YOU."
410   @ : @"Be sure to turn the Locking Tab on the Drive."
420   @ : @"press any key when you wish to continue...."; : Get\0\I$(0,0)
430   On Error Goto Erase'2 : Erase"error"
440 *Erase'2 : On Error Stop
445   On Error Goto Init'disk : Erase"error2"
450 *Init'disk : On Error Stop
460   Sh"initflop sfda >* error"
470 *Make'file'sys : Sh"makfs sfda >* error2"
480   Open\1,11\"error"
490   On Error Goto Open'2 : Get\1,0\Error$ : Close\1\ : On Error Stop
500   Goto Trouble'with'init
510 *Open'2 : On Error Stop : Close\1\
520   Open\1,11\"error2"
525   On Error Goto Successful'init
530   Get\1,0\Error$ : On Error Stop : Close\1\
535   Goto Trouble'with'makfs
540 *Successful'init : On Error Stop : Close\1\
550   Sh"mount sfda /aa"
560   Create"/aa/label.dat"
570   Open\1,10\"/aa/label.dat"
590   Put\1,0\Label$(-1)
600   Close
605   Sh"unmount sfda"
610   Gosub Crt'clear
620   R=10 : C=5 : Gosub Loc : @"The ";Label$;" Disk Has Been Inited"
630   @ : @"press any key to continue..."; : Get\0\I$(0,0)
640   Run
650 *Trouble'with'init : If Init'try=4 Then Gosub Crt'clear
660   Init'try=Init'try+1 : If Init'try<4 Then Erase"error2:Goto init'disk"
670   If Init'try=5 Then Problem$="C-74" : Goto Big'trouble
680   R=10 : C=5 : Gosub Loc : @"The INIT was not successful"
690   @ : @ Tab(4);"Remove the Disk, and insert another New Disk"
700   @ : @"press any key to continue..."; : Get\0\I$(0,0)
710   Goto Init'disk
720 *Trouble'with'makfs : If Makfs'try=4 Then Gosub Crt'clear
730   Makfs'try=Makfs'try+1
735   If Makfs'try<4 Then Erase"error2:Goto Make'file'sys"
740   If Makfs'try=5 Then Problem$="C-99" : Goto Big'trouble
750   R=10 : C=5 : Gosub Loc
755   @"The Attemp to Make a File System  was not successful"
760   @ : @ Tab(4);"Remove the Disk, and insert another New Disk"
770   @ : @"press any key to continue..."; : Get\0\I$(0,0)
780   Goto Make'file'sys
790 *Big'trouble : Gosub Crt'clear
800   R=10 : C=5 : Gosub Loc
810   @"We have encountered a problem which requires additional help"
820   @ : @"Call the USER HOT LINE NUMBER and report your problem"
830   @ : @"PROBLEM NUMBER ";Problem$
840   @ : @"press any key to continue..."; : Get\0\I$(0,0)
850   File$="/sysman/trouble/newdisk"+Date$("")+Time$("")
860   Open\1\File$
870   Put\1\Problem$(-1)
880   Close
890   Run"menu1.sys"
4000 *Crt'clear : @ Chr$(27);"+"; : Return
4010 *Loc : @ Tab(0);"": : @ Chr$(27);"a";R;"R";C;"C"; : Return
5000 *Sav : Save"newdisk.sys"
```

For those who may not be familiar with my programming techniques, I make extensive use of Line Labels. I will always say **Gosub Crt'clear**, for example, and never **Gosub 4000**.

The Line Label, **Sav**, is a technique I use to prevent overwriting a file in error.

The Subroutine, Loc, which is used to address the cursor, is peculiar to the Wyse terminal. If you use this program, you will have to change this to suit your terminal.

Line 100 is a technique used to fill a string with spaces, in this case forty, so that the variable can be used to delete a line on the screen, as in Line 300. For those who may not be aware, terminating a print statement with a semicolon prevents the cursor from "bouncing" around on the screen.

In Line 310, the use of the Valc Command insures that the User has input a number, and Line 320 insures that the number is greater than or equal to 1 and less than or equal to 6.

Line 350 is a Data Statement, and Line 360 uses the Repeat Until Command to assign a value to the String Variable, Label$. If the User's choice is 3, then Label$ will become "Sales", since it is third in the Data Statement.

Lines 460 and 470 use the Sh Command to pass the Initflop and Makfs Commands to Cromix. The >* error portion of the Command will create a file named "error" and redirect any error messages to it. Therefore, if we open that file and get an error when we try to read it, as in Lines 525 and 530, we know that the procedure to initialize the floppy was successful. This same logic is duplicated with the Makfs Command. If the Initflop and Makfs Commands are successful then Line 550 uses the Sh Command to mount the Floppy as directory /aa. Lines 560 through 600 creates and writes Label$ to the file named label. Line 605 uses the Sh Command to unmount the Floppy.

The Routine to actually backup the directory is included in another Basic program. A portion of this program follows.

Line 500 mounts the Floppy. Lines 510 through 540 open the Label file and compare it to the variable Label$ to insure that the correct Floppy has been inserted in the drive.

```
100 Dim Command$(49)
500 sh"mount sfda /aa"
510 Open\1,10\"/aa/label"
520 Get\1,0\Floppy'label$
530 Close\1\
540 If Floppy'label$<>Label$ then Goto Wrong'disk
550 Command$="cptree -vt /data/";label$;" /aa"
560 Sh Command$
570 Sh"unmount sfda"
580 Goto What'Next
```

Line 550 demonstrates that the string used with the Sh Command does not have to be literal as in the first program. Here we have concatenated a literal string with the Variable, Label$, to create the Variable, Command$, which we use with the Sh Command.

One problem with backing up to Floppies, particularly small ones, is that Cptree does not calculate beforehand to insure that the entire directory will fit on the floppy. By using the L Command and redirecting the output to a file, I have developed a Basic Program to calculate the portions of a directory which will fit on a Floppy, Cptree- ing those, instructing the User to remove the Disk and Insert another, which is labeled as the 2nd of that directory. As they become necessary the 3rd, 4th, etc. are added. I will include this in a future article, but if you wish to have it before then, please write, and I will get you a copy of the program.

I would like to thank those who have written concerning the articles I have contributed. If anyone has any questions or suggestions, please write. My address is **PO Box 50119, Nashville, Tennessee 37205.**

Cromemco 3102 Terminal Emulator for the IBM PC

With full file transfer capability

ProCall/PC *plus*

from the longest name in Cromemco Software:
ProtoMatrix Software Development

ProCall/PC *plus* for the IBM PC (and compatibles) is the only program that offers you all these features:

- Complete 3102 terminal emulation, including video attributes, editing, and graphics characters.
- DEC VT100 and VT52 emulations also included.
- Supports file transfers with other computers using these major protocols:
  Xmodem PcXfer
  Kermit Rfile/Sfile
- All assembly language.
- User friendly pop-up menus and dialog boxes.
- Speeds from 110 to 19,200 and 38,400bps.
- Also an excellent general purpose communications program. Use it with BBS systems, telex via Easy-Link, news and information services, and more.
- Ability to transfer files with S100 Cromix, Cromix plus, CDOS, C-10 and many other systems.
- Answer mode for remote operation.
- Easy-to-read manual.

And much more, all for under $250.00 (+ tax). ProCall/PC (without emulation) is under $150.00

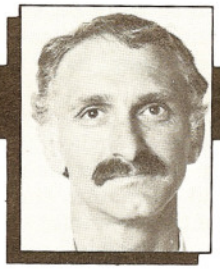
For more information on ProCall/PC *plus* and other products from PSD, call your nearest Cromemco dealer or PSD at: (408) 263-8665

Or write to us at:
1145 Park Heights Drive
Milpitas, CA 95035
USA.
Telex # 503898

PSD

back issues

of I/O News are available



C-10 ENCOUNTERS

C-10 ENCOUNTERS is a regular column directed to users of Cromemco's personal computer, the C-10. It is edited by Dr. Tom Beer, of Applied Environmetrics, located at 118 Gordon St., Balwyn, Victoria 3103, Australia. Dr. Beer can be reached by phone during business hours at (03) 817-2571. Submit editorial directly to Dr. Beer.

Terminals and Oddments

In the very first C-10 ENCOUNTERS column in the Jan/Feb 1984 issue of *I/O News (Vol. 3 #6)*, I promised that in future columns I would deal with the weird and wonderful things that one can do with the C-10 terminal. Since then, choice morsels concerning screen clearing, turning the cursor on and off, altering the brightness and homing the cursor have appeared, but I have never got around to a systematic discussion of the C-10 terminal. One reason is that it is hard to make the subject interesting. A list of the most useful and common Escape codes as given in **Table 1** is incredibly useful and incredibly boring.

Table 1: Some useful terminal Escape codes (The given letter is preceded by Escape)

E	to clear the screen
H	to home the cursor (move it to the top left of the screen)
q	turn off cursor
r	turn on cursor
Z	toggle cursor
F	to precede a cursor address sequence
Y	the same as Esc F
d@	set terminal back to normal mode

(Source: **Cromemco C-10 Technical Manual Appendices)**

Two things have made me realize how mysterious the terminal can be. I have acquired an IBM compatible system and have found it very difficult to get information on the sequences that are supposed to control the terminal under MS-DOS. On an MS-DOS systems disk there is a file called ANSI.SYS that contains the American National Standards Institute codes for terminal operation. Not one of the six books about the IBM-PC that I have consulted bothers to list these "standards." Thus I sympathize with all those C-10 owners who have no idea what the terminal codes are for their machine.

The second event that led to my increased enlightenment was a determined effort to get the CP/M family history package, **ROOTS/M**, running on the C-10. In order to cater to different terminals, ROOTS/M has a file called cod.nrm (normal console driver) which contains the codes to clear the screen, home the cursor, and so on. Because these are different for each computer the designers of ROOTS/M provide an installation program that asks you to enter in the codes for your own terminal. I got a working cod.nrm file eventually. But it was not easy.

The Escape sequences shown in **Table 1** are designed to work when sent to the terminal by a program. To test them out it is possible to put the terminal in local mode and then issue them. For example, to draw a rectangle on the screen try this:

- Control-Shift-S (Press Control, keep it down, press shift, keep both down, press S). Displays status line.
- Control-Shift-L (You are now in local mode.)
- ESC E (Press the ESC key, release it, then press capital E)
- ESC R (Turns on Graphics mode)
- @\\\\\\D (Draws top of the rectangle)
- ESC B (Moves cursor down a line)
- d RETURN d ESC B RETURN (Press the six keys in turn)
- H\\\\\\L (Completes the rectangle)
- ESC S (Graphics mode off)
- Control-Shift-L (Back to online mode)

If you make a mistake then ESC M will delete the line. Notice that capital letters and small letters behave differently. Do not confuse the two.

Terminal Installation

I believe that a large number of commercial packages come with a terminal installation routine. I have been fortunate that in most cases the packages that I obtain have already been preconfigured for the C-10 or its equivalent. Equivalent configurations that should work on the C-10 include the Cromemco 3102 terminal, or the Beehive ADM series. If, however, you have to do it yourself you will need the C-10 technical manual. The terminal codes are not given in the user manual supplied with the machine.

You would also be well advised to have a printer. I found it vital to be able to check a printout of the questions that the installation program asked, along with my answers. The ROOTS/M install program started by asking questions about the input to be expected from the keyboard. Do the arrow keys generate two character sequences? No. What is the hexadecimal value for cursor-up. My reading of Appendix F of the technical manual is that 8B is sent. However if one presses the up-arrow key in response to the CDOS prompt then ²K appears on the screen. This is the screen representation of Control-K, or 0B, and it appears that CDOS may be stripping the high bit so as to convert 8B to 0B. The cursor codes that seemed to work were 0B, 0A, 08 and 0C for up, down, left and right respectively.

The CKBA keyboard does not have a HOME key, whose function is to move the cursor to the top left corner of the screen. After a lot of experimentation I finally discovered that the installation program would not work properly unless one answered something for the cursor-home code. I finally answered 09, (Control-I), which is the code for the TAB key. This means that in this program the TAB key acts like a HOME key. Finally I informed the enquiring program that the explicit DELETE key generates the code 7F.

Next, the installation program asked questions about how the main program should control the visual display. It started by

CROMEMCO SYSTEM III

3102 Term., 3355A Printer, 20 Mb HD,
4 x 299A/B Drives, 5x64K Memory,
TU-ART, PRI, 16 FDC, etc.
CM5A (Beehive) Term., Z-2H HDD,
Assembly, 11.27 Cromix, Software,
Mounting Cabinet, User/Tec Manuals.

$3500 or Best Offer

Call (707)965-2053
Will Sell Major Components Separately

wanting a string to re-initialize the console upon exiting the main program. I gave it 1B 45 1B 72 1B 64 40. This is usually written Esc E Esc r Esc d@ and represents three codes to respectively blank the screen (Esc E), turn on the cursor (Esc r) and enter normal video attributes (Esc d@). If this puzzles you then consult the following sections of the technical manual: **Appendix H** for the conversion between the hexadecimal and Ascii character codes, **Appendix E** for the Escape codes and **Appendix C** for the video attribute selections. Strings for cursor—home, forwards, backwards, up and down are 1B 48, 1B 43, 1B 44, 1B 41, 1B 42 respectively. These are Escape followed by H, C, D, A and B respectively. For more information on their operation read Chapter 2 of the technical manual. A string to erase from the present cursor position to the end of the current line is 1B 4B (Esc K).

Video attributes became slightly more vexatious. The C-10 supports video attributes such as reverse video, blinking screens, underlining, etc. with certain limitations. The Escape code governing a video attribute applies from the position in which it is given (i.e. the C-10 terminal does not leave a blank character after selecting an attribute) until it meets another video attribute downscreen. However, only 16 attributes can exist on one line—a limitation that can become onerous in Writemaster. If eight words on a line are underlined then the underlining will continue to the end of the line whether you want it to or not. It will print out fine but the screen cannot handle more than 16 video setting changes per line. Interesting that in *I/O News Vol. 5 #5*, Dr. Jeff Johnson, who used to be in charge of Cromemco software development writes "...Don't expect users to understand ... that they can't underline more than five words on one line..." With Writemaster you can't underline more than eight words on one line. As this is a hardware restriction there is probably very little that Dr. Johnson could have done to overcome it.

Setting reverse video is done with 1B 64 50 (Esc dP), and I tried to reset it with 1B 64 40 (Esc d@). (This is not quite correct, and I will come back to this point after another two paragraphs). Unfortunately the result turned out to be quite unpleasant. The program seemed to assume that issuing the reset code would turn off all video attributes. On the C-10 the reset code has to be given with the cursor at the exact location where the code to set the attribute was given. This meant that various screens would leave remnant patches of reverse video when the program was finally run. In the end I found that I preferred installing the program without any video attributes.

The C-10 will not support an underline cursor. Codes to turn the cursor on and off are 1B 72 (Esc r) and 1B 71 (Esc q) respectively. Addressing the C-10 cursor is done by sending two characters after a 1B 46 (Esc F). The computer subtracts 20h from each of the two characters and interprets the first as a line number (0 is the top line) and the second as a column number. No intermediate or ending strings are needed.

Finally, there followed various questions on block graphics. In order to be able to draw family trees on the screen the program needed to know the codes for line segments that make up the corners and sides of a square. For the upper-left corner this was 1B 52 40 1B 53, for the upper-right corner it was 1B 52 44 1B 53, for the lower left 1B 52 48 1B 53, for a horizontal line segment 1B 52 60 1B 53, and for a vertical line segment 1B 52 64 1B 53. In each of these cases 1B 52 (Esc R) turns graphics mode on and 1B 53 (Esc S) turns it off. As you have already practiced drawing a rectangle, you may like to check that the above codes agree with the rectangle drawing example.

The C-10 has a graphics mode—which is the mode that we used to draw the rectangle, and a graphics character set. A demonstration of the contents of this character set, along with the other three character sets, is provided by the program CHARSETS on system disks prior to Release 5 CDOS. To experiment with the graphics character set put your machine in local mode and press the three keys ESC d D one after the other. Start pressing keys. To return to the standard character set press ESC d @. Miss K. Baulch wrote to me about installing Multiplan on the C-10 and pointed out that the graphics character set cannot be used to make a square box. She also pointed out that when turning off reverse video, one has to include an Esc e (to turn off reverse video) as well as an Esc d @ (to return to the normal character set). I thought that Miss Baulch's point may help me to get the highlighting on ROOTS/M working properly but, alas, it did not do any better.

Miss Baulch also tried to install **Infostar**. She says that there are two install programs—one for **Datastar** and another for **Reportstar**. The Datastar install program had a Beehive 150/Cromemco 3100 terminal on their menu and she chose that one, but this terminal is not included in the Reportstar installation. She sent her Infostar disk with the two installation programs on it along with a copy of the section of the manual for the program that dealt with installation. This allowed me to make her a new Reportstar program with the same terminal installation as she was using on Datastar. To do this I examined both programs using debug and noticed that the code in Appendix C of the Datastar manual (which tells you where the terminal installation codes appear in the Datastar program) appeared to lie at the same location in the program report.com. I thus transferred across the information from offset 140H to 1A7H from the Datastar program and inserted it into the Report program.

Odds & Sods

1. Miss Baulch also wrote up her installation instructions for **Multiplan** for the Cromemco C-10, and if any reader needs it I will be pleased to send them a copy.

2. A number of readers have asked for more information about the **Baber 5.25 inch conversion unit** mentioned in the column in *I/O News Vol. 4 #5*. The unit is built by Baber Enterprises Pty. Ltd. of 2 Monomeeth Drive, Mitcham, Vic. 3132 Australia. Telex AA35462. It will accept input from a large variety of floppy disks, including the C-10, and output to a large variety of typesetting systems.

3. Some readers have misinterpreted my comments in *I/O NEWS, Vol. V, No. 4*, concerning disk transfers between the C-10 and the IBM-PC. The Cromemco program Diskmaster can be used to do this, even when Diskmaster only supports PCDOS 1, because PCDOS 2 checks the disk type in order to work out how to read it. Thus a PCDOS 1 disk can easily be read by PCDOS 2. However, as the original version of Diskmaster (DIMR-1) only knew about PCDOS 1, any transfer between PCDOS 2 and the C-10 (or vice versa) must use the program Diskmaster along with a disk that was formatted on the IBM-PC using the /8 switch. This disk will work fine on the IBM-PC.

The updated version of Diskmaster (DIMR-2) seems to support both PCDOS 1 and PCDOS 2 (though the documentation does not explicitly say this). In fact, Art Burrows of Tensor Pty. Ltd. writes and tells me that it also works fine with DOS3.1! It would still be nice if Diskmaster allowed transfer to and from Cromix disks.

WE'VE MOVED!

Our new mailing address is:

The I.A.C.U. - I/O NEWS
24843 Del Prado, Suite 473
Dana Point, CA 92629-2852
(714) 661-9764

INSIDE CROMIX

INSIDE CROMIX is an open forum on both eight-bit and 16-bit versions of Cromix. The subject matter is directed towards helping Cromix users derive more from their systems. Members' contributions are invited. INSIDE CROMIX is edited by Jordan Siedband, who can be reached at 5017 Fairview Lane, Skokie, IL 60077, (312)674-1175.

INFORMIX— A Super Data Base Manager

Most computer applications for this consultant have been in number-crunching and data base management. Probably, one of the best data base management systems that I have used is INFORMIX. This package is available for MSDOS, CROMIX and UNIX. Files are compatible among all three systems. It can be used as a complete free-standing data base entry system and report writer, or as a set of 'C' library routines integrated into 'C' programs. It is this feature which, I, as a programmer, am mainly interested. The free-standing system will be discussed in a future article.

I/O NEWS serves the needs of a variety of readers and it is possible that there is an isolated person who does not know the general idea of data base management. Any of the "old-time" Cromemco users who bought the old Data Base Management System and Data Base Reporter learned to arrange their data bases in some intelligent order. However, that package was written in BASIC and was very slow.

The purpose of a Data Base Management System (DBMS) is to organize a set of related data into some pattern for easy entry, editing and retrieval. To this end, some thought must be made by the data base user. The initial step for the user is the SCHEMA or data base file pattern. Suppose we would like a simple mailing list. First, name your data base, e.g., mailing, and then your file, e.g., mlist. Now let us arrange the necessary information in a general pattern.

```
database mailing
file mlist
field clientno    type serial                 index nodups
field lname       type char     length 17     index dups
field fname       type char     length 17
field address     type char     length 17
field city        type char     length 15
field state       type char     length 3
field zip         type char     length 11
field phone       type char     length 13
end
```

A serial type variable is one generated by INFORMIX (of type "long") in serial order. It may be set to start at any numerical value. It is unique for each entry (the "nodups" specifier) and so may be used for check numbers, employee numbers, invoices, etc.

Note the character strings (type char). All lengths are odd if you are going to use these in 'C' programs but not necessarily for free-standing applications. In the programs, these wil be replaced with strings whose lengths are greater by one, to cover the final null at the end of the string. This is because string types in 'C' must lie on storage boundaries. That is, strings in programs will always be even in length. Other types besides char are integer, double, long, float, serial, money, date, and composite. There are several date forms, but I heartily discourage their use since they do not operate correctly for dates past 1999. I have a complete set of date entry and storage routines written in 'C' which will work up to the year 2100. I do not expect to be here then! Also, notice that the key is alphabetically by last name. Should we desire to have storage by last, then first, introduce another field:

```
field name        type comp     lname,fname index nodups
```

We have now introduced a key in addition to the original which indexes the original file by last name and first name, not permitting any duplicates. Comp indicates that it is a composite key. The file does not alphabetize, it gives the address of the entries by key.

The next step is to convert this schema into a viable data base. This is accomplished with the use of a program called DBBUILD. This program will create the data base and create several files for your convenience. A file mailing.dbd stores the variable information in the data base dictionary. The file mlist is saved as mlist.dat, mlist.idx, and perhaps mlist.lok for CROMIX. These steps are identical for free-standing data bases.

At this time, let us introduce another wrinkle. Suppose that I write invoices for my clients. It is not necessary to save the name and address since I might refer to him by clientno and "look up" the corresponding address information. Make a schema as follows:

```
database mailing
file billing
field clentno     type long                   index dups
field date        type long
field invno       type serial                 index nodups
field amount      type double
field id          type comp     clentno,invno index
end
```

Notice that the variable clentno MUST be different from the clientno in the mlist file. This will be used in the sense of a relational data base in which files may refer to keys in other files. Also, since I may have more than one invoice for a given customer, I must permit duplicates in the key.

The data base could be maintained manually by use of the program called Informix, but the use of the ALL (Applications Language Library) for the 'C' language gives me more flexibility because compound keys may be used.

In a 'C' program, the database and its attendant files must be opened and keys specified before they are used. The function dbselect is the one which has options to open or close a database and its files. The files themselves are described by appropriate structures.

Once a given data base has been constructed, there are several important considerations. Information must be added, altered, or deleted. Also, the data must be retrieved in some specified order. A mailing list may be retrieved in alphabetical order, ordered by zip code, or even a compound key such as alphabetically by zip. Fortunately, ALL has a rich collection of options to browse through the data base. If you give the value of the key you wish, dbfind permits COMPARISON, EQUAL, GTEQ, GREATER, FIRST, LAST, CURRENT, NEXT or PREVIOUS. With these options, and some of my own wits, I can pin-point almost any item even in a very large data base.

The functions dbadd, dbupdate, dbdelete, have obvious uses for specific records. Should a specific key be chosen, and the user has located the first set of that key, the user may get the

next or previous item by key and so print out all or selected parts of the data base. Thus, at the end of the month, statements for all clients may be prepared alphabetically by zip code or any other sort of your choice. If you did not make all of the keys at the start, INFORMIX gives you the freedom to add or delete keys in the data base so that the current data base reflects your choices.

There are several other DBMS's available for CROMIX-PLUS or UNIX, but in my opinion, none can match the richness and variety of INFORMIX. Some of my clients use the full power of INFORMIX with the complete report writing package, ACE, but I feel that by using ALL, I have complete control. I can add items, delete them, or update them using C programs at a significant fraction of the time anyone else uses. One of my clients insisted on buying an IBM-PC and then tried to manage a data base without the necessary discipline. He had steadfastly refused to buy a CROMEMCO because it was not IBM compatible. Now he has a little box in the corner which nobody programs or uses, since the machine has not been tailored for the application. Although INFORMIX and CROMEMCO have not made life perfect, it is certainly more pleasant! CD

System 250

Continued from front cover

Chris Mukai of Cromemco demonstrates the MegaSafe™ removable hard disk in the new System 250 computer.

capacity of Cromemco's current CTD tape drive, and will be offered in other Cromemco systems later this year.

The main markets that Cromemco has targeted for the System 250 are applications that require high levels of data security. Since the hard disk is completely removable, sensitive or classified data can be completely removed from the computer when required, and stored securely away. Quick exchange of the hard disk also permits ready change of the computer environment when required. The machine can be switched from an operational environment, for example, to a test or developmental environment without any possibility of destroying or altering operational data during test or development use.

Like other Cromemco XXU-based computers, the System 250 can operate under either the Cromix Plus operating system or under the UNIX System V operating system. A full range of development and application software is available including the C, Pascal, Fortran, and Basic languages and the Informix, Unify, and Today databases and 4th generation languages.

Technical data on the System 250 is given in the accompanying table, and further information is available directly from Cromemco.

FIGURE 1

Table—System 250 Specifications

PROCESSOR	Motorola 68020
CO-PROCESSOR	Motorola 68881
RAM CAPACITY	2-16 megabytes (ECC available)
SERIAL CHANNELS	9-65 channels
DATA EXCHANGE	5" floppy disk
DATA BACKUP	60 megabyte tape drive
HARD DISK CAPACITY	190, 380, or 760 Mb (removable)
MOUNTING CONFIGURATION	19" rack mount

CD

CROMEMCO SYSTEM III

11 Mb HDD Hard Disk
2 - 8" DS/DD Floppy Drives

Boards
- 1 — ZPU
- 4 — 64KZ
- 1 — 256KZ
- 1 — WDI II
- 3 — TUART
- 1 — PRI
- 1 — 16 FDC
- 1 — 4 FDC

Terminals & Printers
- 3 — Cromemco 3102 Terminal
- 2 — Beehive DM5A Terminal
- 1 — Racal-Vadic Acoustic Coupler
- 1 — Cromemco 3355A Printer
- 1 — G.E. Terminet 200 Printer

Misc. Cabling and Connectors

Software
- Cromix Operating System
- DBMS
- Accounting System
- Structured Basic

$5000 or Best Offer
Call Chris Weierman (918) 825-2966

FOR SALE

SMD controller - Cromemco SMDI-X	$700
SMD controller - Konan SMC-200	$350
DPU	$600

above running under Cromix-Plus series 150. Cromix-Plus now includes SMD drivers - no extra software is required.

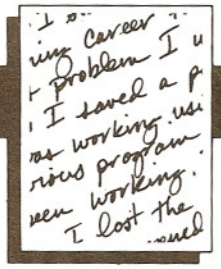
ALSO:

Tuart or FDC to DB25 cables (36") ea	$10
64KZ ram (complete but not working)	$20
4FDC (complete but untested)	$40
ZPU	$100

ALSO:

Sanders Technology Media 12/7 typographic printer with sheet feeder, many fonts & Spellbinder software, works great. $1500

Call Gary
714-978-7860 days 631-0524 eves



USER NOTES

USER NOTES are useful techniques, tips, ideas and other helpful information gleaned from our member's experiences with their Cromemco systems. If you have something that you would like to share, write it up and send it to I/O NEWS, c/o USER NOTES, 24843 Del Prado, Suite 473, Dana Point, CA 92629-2852.

UNIX Flush Program

Editor's Note: *The following UNIX System V flush utility was contributed by Bernd Felsche, B.E. of Riverton Engineering Co. Pty. Ltd., 41 Austin Avenue, Kenwick, Western Australia 6107.*

Just in case you have not already seen anything like this before, I would like to share the following program with other members of the IACU who have a Cromemco machine running UNIX System V.

The program is written in C and replaces the Cromemco-supplied shell script /usr/bin/flush.

The reason for replacing the shell script is mainly to reduce system overheads during normal running (especially) if the large kernel is used.

The shell script as supplied by Cromemco with their UNIX releases has the disadvantages of larger size and slower execution speeds when compared to the program entirely coded in C.

The main speed advantage is obtained by the following mechanisms:

1) Smaller code size means that the process is less likely to be swapped out under normal system operation.
2) Programs sleep and sync do not have to be loaded into memory every two minutes, either from the swap area, or from the /bin directory.
3) Program initialization only takes place once when the program is started on going to multi-user. The shell script requires the initialization to take place twice every two minutes.
4) Extra processes do not have to be created by system fork calls which the shell procedure requires to start the sync and sleep programs.
5) Instructions are in machine executable form and do not have to be interpreted.

The total CPU time spent executing the shell script used to be about 5 minutes in every 24 hour period on our system. The flush program in C takes about 30 to 45 seconds over the same period.

Two other advantages are coincidental but by no means trivial:

1) Every time the flush shell script executes the sync-sleep sequence, two more processes are created. If accounting is turned on, then the file /usr/adm/pacct must be updated to show this. The disk access time which this eventually requires further degrades system throughput. As the /usr/adm/pacct file grows, it requires more disk space.
   At the end of the day when the process accounting summaries are created, roughly 1400 more records (for two processes, once every two minutes) in the /usr/adm/pacct file must be processed. Hence, the system process accounting takes longer.
   The summary files created by the process accounting will also be marginally smaller and hence require less of your precious disk space.
2) The process table will clearly show the flush program and it will only occupy one slot at all times, whereas the process table has to change to arrange the sync and sleep programs to run via the fork and exec system calls made by the shell when it executes the flush shell script.

All of the aforementioned reasons may appear to be rather esoteric and the benefits minimal. However, the small amount of work required to enhance system performance seems well worth it in the long run.

To get the C program to run, it must first be entered as a file called flush.c using your favorite editor in a non-system directory (e.g. /usr/joe). The program must be then compiled using the command

```
cc -o flush flush.c
```

which will create the executable program. You may wish to strip the symbol table from the program using the strip command but this is not essential.

Next you must use superuser privileges to move the program to the /usr/bin directory and change the ownership of the file to root. Save the Cromemco distributed shell script just in case you blow it.

```
mv /usr/bin/flush /usr/bin/flush.cro
mv flush /usr/bin
cd /usr/bin
chown root flush
```

The program is now ready to run except that the file modes must be changed to protect the program and to ensure that it can function fully:

```
chmod 4755 flush
```

This assigns read and execute permissions for all users and write permissions to the owner (root) only. The setuid bit is set to permit the program to update the disk fully.

That's about it. You can check the operation of the program by shutting down the system to single user and entering:

```
flush 120 &
```

The program must be run in background, otherwise it will hog your terminal as it catches hangup, interrupt and quit signals.

Copy a large file to a temporary directory and see if the buffers are flushed...

```
cd /tmp
cp /unix joe
cp /unix fred
sleep 130 ; – to allow flush to work
sync
sync
```

If your shell prompt returns almost as rapidly from the first sync command as it does from the second then the flush program is working.

The flush program will be used instead of the shell script next time you go to multi-user if the multi-user section of the /etc/rc file contains the line:

```
/usr/bin/flush 120 &
```

Note that I have stuck to the standard 120 second interval between buffer flushes. You may reduce or increase this, depending on your degree of paranoia about file system integri-

ty if you have an unexpected system failure.

You may notice if you have the flush program running on a system with no active users, (and you use the system activity package) sar may indicate that the system is 100% idle! This is normal and indicates that less than half of one percent of system time is spent on overheads.

P.S.: I am working on a suite of programs and procedures to monitor printer usage. I will send you a copy and documentation as soon as it is tested and documented.

```
/*                      flush utility - by Bernd Felsche
                Riverton Engineering, Kenwick, Western Australia

        This replaces the shell script (/usr/bin/flush) distributed
        by Cromemco with their UNIX releases.

*/

#include        <stdio.h>
#include        <signal.h>
#include        <ctype.h>

/*      define infinite loop            */
#define FOREVER 1

main(argc,argv)

int argc;
char *argv[];
{
        int c,j;
        unsigned seconds;

        if (argc != 2){
                fprintf(stderr,"%s: Usage: %s interval\n",argv[0],argv[0]);
                exit(1);
        }
        for (j=0;argv[1][j];j++){
                if (!isdigit(argv[1][j])){
                        fprintf(stderr,"%s: Numeric interval required\n",argv[0]);
                        exit(2);
                }
        }
        seconds = atoi(argv[1]);
        if (seconds){           /*      ignore external signals */
                signal(SIGHUP,SIG_IGN);
                signal(SIGINT,SIG_IGN);
                signal(SIGQUIT,SIG_IGN);

                while (FOREVER){/*      loop around and flush buffers */
                        sync() ;
                        c=sleep(seconds);
                }
        }
        else {
                fprintf(stderr,"%s: Must have non-zero interval\n",argv[0]);
                exit(3);
        }
}
```

Clist

Editor's Note: *The following note and listing were contributed by Colin "Soup" Campbell, of Soup's Electronics, P.O. Box 81871, College Branch, Fairbanks, AK 99708. Telephone: (907) 456-3487.*

I have written this utility for in-house use because I needed both a record of listing date and file status, but also indentation that I could vary and still keep functions on narrow paper even when nesting gets deep. Then as all things evolve, counting the levels of indentation seemed easier with -c and an output file -o seemed to be appropriate. I thought this a simple utility when someone else said why not share it with others. So in the spirit of *I/O NEWS* and user groups ... here is a listing. Happy computing!

```
/* clist program started 12/10/86 by soup     */
/* this program is put into the public domain */
/* please pass it on to others without charge */
/* and give credit to where credit is due. If */
/* you make major enhancements send me a copy */
/* so that I might benifit from your insight. */
/*                                            */
/*              Soup's Electronics            */
/*              P.O. Box 81871                */
/*              Fairbanks, Alaska 99808       */
/**********************************************/

#include <stdio.h>
#include <jsysequ.h>

int     verbose=0;
int     pause=-1;
int     offset=0;
char    buf1[256];
char    buf2[256];
char    in_file[256];
char    outfile[256];
char    dummy='0';
char    in_char=' ';
int     in_amt=5;
int     page_len=23;
struct  sys_time *time;
struct  sys_date *date;
struct  st_time *t_create;
struct  st_time *modify;
FILE    *out_file=stdout;

main(argc, argv)
int     argc;
char    *argv[];
{
extern  FILE    *out_file;
        int     of;
        int     fd;
extern  char    buf1[];
extern  char    buf2[];
        int     i;
        int     in=0;   /* linecount of input file */
        int     line=0; /* linecount of output page */
        char    *s;
        char    ch;
if (argc<2)             /* file to list is necessary */
        usage();
while (--argc > 1){
        ++argv;
        if (*argv[0]=='-'){
                s=argv[0]+1;
                switch (*s){
                        case 'O':       /* output filename */
                        case 'o':
                                of=create(*++argv,op_write|op_condf);
                                if(of==-1){
                                        printf("File already exists!\n");
                                        exit(0);
                                        }
                                close(of);
                                out_file=fopen(*argv,"w");
                                pause=0;
                                argc--;
                                break;
                        case 'V':       /* verbose */
                        case 'v':
                                verbose=-1;
                                argc--;
                                break;
                        case 'L':       /* page length */
                        case 'l':
/*                              printf("pagelength: %d\n",atoi(*++argv));*/
                                page_len=atoi(*++argv);
                                argc--;
                                break;
                        case 'I':       /* indentation */
                        case 'i':
                                in_amt=atoi(*++argv);
                                argc--;
                                break;
                        case 'C':       /* char to indent with */
                        case 'c':
                                strcpy(&in_char,*++argv);
                                argc--;
                                break;
                        default:
                                printf("Option not found %s\n",*argv);
                                usage();
                                break;
                }
        }
}
strcpy(*in_file,*++argv);
if(((fd=open(*in_file,"r"))==-1){
        printf("Can't open %s\n",*in_file);
        exit(0);
        }
start_up();
while(geti(fd,&buf1[0], 255)){
ln++;
if (line>=page_len||line==0){
        if (pause&&line!=0)
                ch=getchar();
        if (verbose){
                print_title();
                line=2;
                }
        else line=0;
        }
move_buf();
if (verbose)
        fprintf(out_file,"%3d:",in);
for (i=0;i<offset+in_amt;i++)
        fprintf(out_file,"%c",in_char);
fprintf(out_file,"%s",&buf2[0]);
if (buf2[0]=='\0')      /* this takes care of blank lines */
        fprintf(out_file,"\n");
line++;
check_buf();
}
fclose(out_file);
close(fd);
}
move_buf()
{
extern  char    buf1[];
extern  char    buf2[];
        int     i=0;
        int     j=0;

while (isspace(buf1[i]))                        /* ignore spaces, tabs, ect */
        i++;
while ((buf2[j]=buf1[i]) != '\0')
        i++,j++;
}
check_buf()
{
extern  int     offset;
extern  char    buf2[];
static  int     comment;
        int     i=0;

while(buf2[i] != '\0'){
if (buf2[i]=='/'&&buf2[i+1]=='*')
        comment=-1;
if (buf2[i]=='*'&&buf2[i+1]=='/')
        comment=0;
if (!comment)
        switch (buf2[i]){
                case '(':
                case '{':
                        offset++;
                        break;
                case ')':
                case '}':
                        offset--;
                        break;
                default:
                        break;
                }
        i++;
        }
}
start_up()
{
date=alloc(sizeof(struct sys_date));
getdate(date);
```

```
time=alloc(sizeof(struct sys_time));
gettime(time);
t_create=alloc(sizeof(struct st_time));
fstat(*in_file,st_tcreate,t_create);
modify=alloc(sizeof(struct st_time));
fstat(*in_file,st_tmodify,modify);
}
print_title()
{
fprintf(out_file,"Printed            Created             Modified   File: %s\n",*in_file);
fprintf(out_file,"%2d:%2d:%2d %2d/%2d/%2d   %2d:%2d:%2d %2d/%2d/%2d   %2d:%2d:%2d %2d/%2d/%2d\n",
        time->hour,time->minute,time->second,
        date->month,date->day,date->year,
        t_create->hour,t_create->minute,t_create->second,
        t_create->month,t_create->day,t_create->year,
        modify->hour,modify->minute,modify->second,
        modify->month,modify->day,modify->year);

}
#include "atoi.c"
usage()
{
printf("Usage: Clist [-c c] [-i #] [-l #] [-o outfile] filename\n");
exit(0);
}
```

Turbo Modula-2 Patches

Editor's Note: *The following patches were contributed by Alberic Muller, Rue du Jura 12, CH-2525 Le Landeron, Switzerland.*

```
        TITLE   M2COMPAT
;       Patch for Turbo Modula-2 V1.00
;       ******************************
;
; To patch V1.00 of Turbo Modula-2 perform following
; step:
;
; * Assemble the patch source in hex format:
;
;   $ ASMB M2COMPAT.@@Z HEX
;
; * Load DEBUG and type the commands:
;
;   $ DEBUG M2.COM        -- Load M2
;   DEBUG version 00.17
;   -FM2COMPAT.HEX        -- Set patch file name
;   -R                    -- Load patch
;   -FM2.COM              -- Set M2 exe file name
;   -W                    -- Write back to disk
;   -^C                   -- Exit to system
;   $
;
; Following problems are patched when executing
; under Cromix-Plus V31.40 and the CDOS simulator
; sim.bin V2.69:
;
; * Get/Set User code is a no operation in the
;   simulator. This means that register A is not
;   modified across the call to the simulator. Reg.
;   A was loaded by the interrupt base vector prior
;   to getting the user code (in fact with a negative
;   number). The patch makes a NOP while setting and
;   always returns 0 when fetching the user code.
;
; * Get Disk Parameters and Get Allocation Vector calls
;   are not supported in sim.bin since the driver data-
;   base in Cromix drivers is much different than from
;   CP/M. To avoid a system abort, these data structures
;   are simulated here. The DSKPAR make a floppy size
;   of 512 KB. The CP/M DMA buffer is simply interpreted
;   as an ALOCVEC.
;
; The patch strategy is to shorten somewhat resident
; M2 error messages, and use the saved space for the
; patches.
;
;
; Define constants
;
ERROR   EQU     0               ;Dummy value
BDOS    EQU     5               ;System entry
GETADR  EQU     1FH             ;Get Disk Parameters
GETALV  EQU     1BH             ;Get Alloc. Vector
GETUSR  EQU     20H             ;Get/Set User Code
DMAADR  EQU     80H             ;DMA address
;
; Define patch areas (for V1.00 only!)
;
MSGBAS  EQU     05B0H           ;Base address of
                                ;runtime messages
BDHOOK  EQU     14E9H           ;BDOS call hook
LOGMSG  EQU     2029H           ;Turbo M2 logon
;
; Macro to check patch consistency,
; (write out message and force assembly error
;  without altering the ALC).
;
CHECK:  MACRO   #BADDR
        IF      #BADDR NE $
        CONMSG  String address mismatch: correct base address
        EQU     ERROR           ;Correct base address
        ENDIF
        MEND
;
        LIST    NOGEN
;
;
; Point to resident error messages
;
        ORG     MSGBAS
;
        DB      8               ;Message size
        DB      'IllInstr'      ;Original message text:
                                ;'IllegalInstruction'
;
; Patch start
;
; Filter out Set/Get User Code and Get Addr CP/M calls
;
PATCH:  PUSH    AF              ;Save accumulator
        LD      A,GETUSR        ;Get call number
        CP      A,C             ;Set User?
        JP      Z,P2            ;If Z yes
        JP      P1              ;Continue
;
; Continue with messages
;
        CHECK   MSGBAS+13H
        DB      12
        DB      'PointerError'
;
        CHECK   MSGBAS+20H
        DB      10
        DB      'BadOverlay'
;
        CHECK   MSGBAS+2BH
        DB      11
        DB      'BoundsError'
;
        CHECK   MSGBAS+37H
        DB      9
        DB      'DivByZero'     ;'DivisionByZero'
;
; Continue with patch
;
P6:     LD      HL,DSKPAR       ;Point to DPB
P5:     POP     AF              ;Restore
        RET                     ;Finish
;
; Continue with messages
;
        CHECK   MSGBAS+46H
        DB      6
        DB      'OutMem'        ;'OutOfMemory'
;
; Continue with patch
;
P4:     LD      HL,DMAADR       ;Point to DMA buffer
                                ;(faked ALV)
        JR      P5              ;Continue
;
; Continue with messages
;
        CHECK   MSGBAS+52H
        DB      14
        DB      'EndOfCoroutine'
;
        CHECK   MSGBAS+61H
        DB      8
        DB      'FunNoRes'              ;'FunctionReturnsNoResult'
;
; Continue with patch and define pseudo CP/M
; disk parameter block
;
DSKPAR: DW      20              ;Sectors per track
        DB      3               ;Block shift,
        DB      7               ;Block mask,
                                ;for a block size of 1024
        DB      0               ;Extent mask
        DW      511             ;Disk storage (0.5 Mbytes)
        DW      127             ;Directory entries (128)
        DB      0F0H            ;Allocation 0
        DB      0               ;Allocation 1
        DW      0               ;Check size
        DW      2               ;Offset
;
; Continue with messages
;
        CHECK   MSGBAS+79H
        DB      7
        DB      'StrLong'       ;'StringTooLong'
;
; Continue with patch
;
P3:     CP      A,C             ;Get alloc. vector?
        JR      Z,P4            ;If Z yes
;
        POP     AF              ;Restore
        JR      P0              ;Drop though CP/M
;
; Continue with messages
;
```

```
173             CHECK   MSGBAS+87H
174             DB      7
175             DB      'RealOverf'     ;'RealOverflow'
176     ;
177     ; Continue with patch
178     ;
179     P2:     POP     AF              ;Restore
180             XOR     A,A             ;Clear User Code
181             RET                     ;Finish
182     ;
183     ; Continue with messages
184     ;
185             CHECK   MSGBAS+94H
186             DB      8
187             DB      'Overflow'
188     ;
189             CHECK   MSGBAS+9DH
190             DB      6
191             DB      'CasErr'        ;'CaseSelectError'
192     ;
193     ; Continue with patch
194     ;
195     P1:     LD      A,GETADR        ;Get call number
196             CP      A,C             ;Get disk par?
197             JR      Z,P6            ;If Z yes
198             LD      A,GETALV        ;Get call number
199             JR      P3              ;Continue
200     ;
201     ; Continue with messages
202     ;
203             CHECK   MSGBAS+0ADH
204             DB      7
205             DB      'BadHeap'
206     ;
207             CHECK   MSGBAS+0B5H
208             DB      10
209             DB      'StackOverf'    ;'StackOverflow'
210     ;
211     ; Continue with patch
212     ;
213     P0:     JP      BDOS            ;Execute system call
214                                     ;(and return to main)
215     ;
216     ;
217     ; Patch to hook all M2 calls to CP/M BDOS
218     ;
219             ORG     BDHOOK
220     ;
221             CALL    PATCH           ;Catch calls
222     ;
223     ;
224     ; Patch Turbo Modula 2 login message
225     ;
226             ORG     LOGMSG
227     ;
228             DB      'Cromix, '      ;ID for patch
229     ;
230             END
```

```
1               TITLE   M2OVRPAT
2       ;       Patch for Turbo Modula-2 V1.00
3       ;       ******************************
4       ;
5       ; To patch V1.00 of Turbo Modula-2 perform following
6       ; step:
7       ;
8       ; * Assemble the patch source in hex format:
9       ;
10      ;   $ ASMB M2OVRPAT.@@Z HEX
11      ;
12      ; * Load DEBUG and type the commands:
13      ;
14      ;   $ DEBUG M2.OVR          -- Load M2 overlay
15      ;   DEBUG version 00.17
16      ;   -FM2OVRPAT.HEX          -- Set patch file name
17      ;   -R                      -- Load patch
18      ;   -FM2.OVR                -- Set M2 ovl file name
19      ;   -W                      -- Write back to disk
20      ;   -^C                     -- Exit to system
21      ;   $
22      ;
23      ; * Patch M2.COM with M2COMPAT.Z80 if not already
24      ;   done.
25      ;
26      ; Following problems are patched when executing
27      ; under Cromix-Plus V31.40 and the CDOS simulator
28      ; sim.bin V2.69:
29      ;
30      ; * Cromix does not support '@' in filenmes. CP/M
31      ;   does however. Turbo Modula-2 uses *.@@@ names
32      ;   in temporary files. Patch the file extension
33      ;   directly in the overlay file.
34      ;
35      ;
36      ; Define patch areas (for V1.00 only!)
37      ;
38      EXTBAS  EQU     5419H           ;Base address of
39                                      ;file extension
40      ;
41      ;
42      ; Patch Turbo Modula temporary file extension
43      ;
44              ORG     EXTBAS
45      ;
46              DB      'M2$'           ;New extension
47      ;
48              END
```

Slave Processor

Continued from front cover

CROMIX, and improvements by several Cromemco dealers have made it run better. With Systems Atalanta's new slave processors, it runs better than ever before. Because of the enhanced capabilities and flexibility it provides, the CROMIX Z80 slave processor may be, for those of us who need it, the greatest advance in computing since the initial introduction of the CROMIX operating system.

With these new CROMIX slave processors, true multiple processing capability is afforded. Each slave processes data or runs applications simultaneously with the main processor and with other slaves. While the power of multiple processors is available, the sophistication and integrated function of a true multi-user CROMIX system is preserved.

Each slave processor consists of a single S100 board with a fast 8MHz, Z80H processor, 256K of RAM, and a high-speed bus interface. Two processes can run in a slave simultaneously, each in a separate 64K bank. Therefore, each slave board can support up to two users, whose terminals may be connected directly to the two serial ports. Users have the option of running applications either in the slave or in the system's memory, using the main processor (e.g. DPU). Practically all Z80 programs that can be run in system memory can also be run in the slave.

In multi-user CROMIX systems, slaves can provide expansibility with an overwhelming improvement in speed and performance. With two users per slave, most systems can accommodate two to four times their present number of users, up to 32. In actual work conditions, the practical number that most systems can support without performance loss logically depends upon the kinds of applications being used. When running applications that are very disk intensive, such as database systems, slaves might permit a system to double its present number of users up to 16.

On the other hand, if applications are primarily processor intensive, using a great deal of processing power but not much disk activity (eg., wordprocessing), then slaves may permit the addition of more than 16 users per system. For most users, disk and processor activity probably balance in their applications, and they could reasonably expect slaves to support at least 16 users with excellent performance.

I/O through the slaves can be handled internally through the serial ports or externally through the S100 bus interface. Therefore, the slave processor board can also serve as a general purpose high-speed serial I/O card. Since the slave needs to work with the main processor only for disk or printer operations, normal console I/O can be managed internally by the slaves and does not go through the main operating system. Since it is independent of the operating system and is not hampered by system overhead, internal I/O is exceedingly fast. Consequently, this option permits users to avoid the processing limits imposed by the usual CROMIX I/O. For example, some users need to collect data from certain types of instrumentation at very rapid rates. The main processor cannot handle data at such rates because of its many tasks, but if the program is run in the slave, the slave processor can collect the data (as much as

40-50K) very quickly and write it to the system for later processing.

However, with applications in which the bus interface must be used, system overhead can be minimized by taking advantage of programs that utilize block I/O rather than single character I/O. Since the typical user has no control over the I/O technique used, some explanation is deserved. Practically all CP/M and CDOS programs use character I/O for the printer unless they have been modified by a competent systems programmer. This means that data is transferred to the printer a single character at a time, which is very inefficient (slow) under CROMIX. Some Cromemco dealers have provided modifications to popular CP/M programs or to the CDOS simulator to transfer data to the printer in blocks of characters. Since little more system overhead is required for transferring a block of characters than for a single character, block I/O provides the most efficient use of system resources, especially when using slave processors. However, caution is advised when using some modified CP/M software. Certain software products, Wordstar as an example, are often modified for use with TUART cards for direct output to the TUART's hardware ports. Such modified software will not run on the slave, since its ports are different from those of the TUART.

Systems Atlanta has developed a utility program, called spl.com, which intercepts printer output from CP/M or CDOS programs and either spools it or sends it directly to the desired printer using block I/O. This utility is available to purchasers of our slave processor for a small handling charge upon return of a completed software license agreement. This permits our customers to use the slave processor most efficiently from the outset. Moreover, our new version of Z80 32K Structured BASIC provides efficient printer I/O and built-in spooling to any printer. Incidentally, all the special features of our new versions of Structured BASIC (*I/O NEWS, Vol. 4, No. 6*) are particularly optimized when running in the slaves. If you've been impressed with our SBASIC alone, you will be amazed at its function with the slave. The two make a powerful combination!

Since the slave processor runs at 8 MHz rather than the usual 4 or 5 MHz of the system processor, almost any program will run faster in the slave. Imagine running an application, such as word processing, with an 8 MHz processor at baud rates up to 38K baud (if the CRT will keep up)! Users are cautioned not to use the highest baud rates unless their program is running in the slave. The slave's demand for data can be so fast that the operating system will virtually lock out all other users while trying to satisfy it.

Depending upon the types of programs and their specific features, slaves can increase the speed of running most programs by two to eight times. For example, when running Wordstar with slaves, as many users as desired can run the programs simultaneously, each having twice the speed as when using the system independently! A software package, such as Wordstar, which typically runs slow in situations of minimal processing power, has greatly improved speed when run in the slave. Also, Systems Atlanta has modified Wordstar to use block I/O for printer output. As a result, whether I/O is internal or external, Wordstar now runs much faster. Many users who have wished to use this package but have avoided it because of speed limitations can now hope to do so with slave processors.

Using dBASE II with the slave processor, comparative times for running a do-while loop counting up to 70 were determined. Operating as a second user on the CROMIX system without the slave (while the first user performed processor-intensive inventory control operations) gave a time of 4.77 seconds for running the loop. However, operating in the slave independent of the system but under like conditions, produced a time of 1.47 for the same operation. Likewise, when using Systems Atlanta's new version of Structured BASIC for CROMIX and our own Inventory Control software package (as the only user on the system), times required for looking up a particular inventory item number were compared. Without the slave, the process required twice as long as when using the slave.

The results from the simple tests shown below illustrate the power as well as some of the possible inefficiencies associated with use of the slave processor.

The times shown in the table are actual test results and include all system overhead required for searching directories, loading programs, and program I/O. All the tests were run under CROMIX 31.05, using Systems Atlanta's Z80 version of Cromemco 32K Structured BASIC (sbasic.bin). Three slave cards were installed with two terminals connected to each. The terminal baud rates were 19,200. The load on the main CPU due to other users was very light. The main processor was a 8Mhz DPU (4Mhz Z80). The system included a large amount of memory as system buffers. Program load times may vary by a second or two, depending upon buffer contents.

Tests A through D show the effects of heavy console I/O as well as the speed of the slave's Z80 processor. The slaves are ideal for handling heavy console I/O when the terminal is connected directly to the slave in which the program runs. There is no load on the main CPU except for the initial program load. The same would be true for any CPU-intensive program.

The slave-system interface through the S100 bus is optimized for block I/O at a sacrifice in character I/O speed. Test D shows the effect of passing heavy character I/O through the system. Note that in this test, each character is passed from the slave in which the program is running to the system, then through the system, and back out to the slave which is attached to the user's terminal. It is clear that programs run in this way should have only a small amount of character I/O compared to the other work to be done.

To determine the time required for console I/O, Test A was repeated with line 10 replaced by 10 j$ = str$(i$). The times were reduced to 1:27 and 1:22, compared with 1.11 and 1.10, respectively, when using the 68000 version of Structured BASIC. Of course, with this variation, there was no change in the time required for the program to run in the slave (1:03 and 0:04).

PERSCI

DISK DRIVE MAINTENANCE
AVERAGE 48 HOUR TURN-AROUND

PPS has been providing solutions for PerSci/Cromemco users, user groups, dealers and OEM's for over 6 years.

With over 43 years of combined technical electronics experience, including 12 years on the PerSci payroll, the experts at PPS are uniquely qualified to provide you with the time conscious results that you require for your highly sophisticated disk drives.

Forget the time consuming multiple warranty returns to accomplish what should have been done right the first time!

All disk drives serviced at our lab are tested for a minimum of four hours on a Cromemco CS-3 to insure operational integrity!

On-site services available in the Los Angeles and San Francisco Bay areas. High volume repairs available on-site internationally.

Also, tandon drive repairs and purchase and resale of new and used Cromemco products.

Call **John Bush**, former Supervisor/Lead Technician of Customer Services at PerSci at **(714) 861-6649**

PPS

PERIPHERAL LABS
547 GREAT BEND
DIAMOND BAR, CA 91765

The speed at which programs and data can be moved between the system and the slave is determined far more by the system overhead than by the speed of the hardware. A block of 256 bytes can be moved from system memory into the slave at a rate of one million bytes per second. At this speed, 30K could be moved in three one-hundredths of a second. Using the ddump utility, such a move actually takes about one second. Loading a 30K program into the slave from a hard disk takes three or four seconds (see Tests E and F). The extra time is required because setting up a program to run in the slave is a complicated process. The slave-run utility, sr.bin, must load and run in system memory. It then must find and open the program file, read it into system memory and transfer it to the slave. Finally, the required communications between CROMIX and the program in the slave must be set up. As a result, getting a program into the slave may take a couple seconds longer but the greater speed of execution in the slave more than makes up for it!

Tests G and H simulate a typical data processing application with disk and console I/O. At user sites, installation of two or three slave processors has consistently shown a four-fold or greater increase in system productivity. The experience of our clients, DeLoach & Company, a CPA firm, provides an excellent example of time savings possible with the slaves. Their office uses a Microsoft BASIC general ledger applications (multi-departmental financial statements for a large company) that formerly required 6 hours to complete. With the addition of 2 slave processors to their system, this application can now be completed in an hour and a half! As mentioned previously, since the system processor operates with a 4-5MHz processor and the slave operates at 8MHz, the addition of two slaves provides four times the usual processing power. As long as the applications are run in the slave and not through the bus interface, processing power increases in simple arithmetic multiples when slaves are added. However, the increase in productivity will drop as the main processor becomes loaded down with serving the slaves.

The cost of a single slave processor with software license is $1095.00, with discounts allowed on quantity purchases. This is a means for users to add considerable processing power to their multi-user systems at a low cost per user. A separate software license is required for each slave in the system. Availability is from stock to thirty days.

Direct inquiries to:
Systems Atlanta, Inc.
P.O Box 99
Lebanon, GA 30146
(404) 928-0240

Test	Command Line	Program Function	Real	CPU
A.	clock -h sbasic test.sav (Run in system memory)	10 for i = 1 to 10000 20 print i 30 next i 40 bye	2:22	2:16
B.	clock -h sr sbasic test.sav (Run in the slave connected to the user's terminal)	same as above	1:03	0:04
C.	clock -h sr -i2 sbasic test.sav (Run in a slave not connected to the terminal and use internal I/O, i.e., send the output to the terminal connected to slave number 2)	same as above	1:03	0:04
D.	clock -h sr -2 sbasic test.sav (External I/O is by default, sending the output back through the system and out to the user's terminal which is connected to slave number 2)	same as above	5:52	5:17
E.	clock sbasic test.sav	10 bye	1.0	1.4
F.	clock sr sbasic test.sav	10 bye	4.0	3.9
G.	clock sbasic test.sav	10 open\1\"sbasic.bin" 20 dim a$(127) 30 on error goto 70 40 get\1\a$ 50 print i,asc(a$),itasc(a$)x 60 goto 40 70 bye	36.0	32.6
H.	clock sr sbasic test.sav	same as above	15.0	7.1

SPECIAL OFFER

COMPLETE YOUR COLLECTION OF I/O NEWS MAGAZINES DURING THIS SPECIAL OFFER

TWO Full Volumes are included in your own I/O NEWS Reference Binder

Valued at $130 Regular Price NOW FOR ONLY $50 EACH!*

Shipping Information: USA: Street address only, no P.O. Box $3.00 ea.
Canada and Mexico: Air Shipping $7.50 ea.
International: Shipping will be invoiced for purchase price and shipping rate. PLEASE SPECIFY AIR OR SURFACE SHIPPING WITH ORDER.

Check One _____ Volumes I and II (Vol. I is in limited supply)
_____ Volumes III and IV

Orders can be paid by check or money order** or by VISA or MasterCard

* California residents add 6% sales tax

** U.S. Dollars. International checks not drawn on U.S. bank add $10

I/O NEWS • 24843 Del Prado Suite 473 • Dana Point, CA 92629 • 714-661-9764

XXU Benchmarks

Continued from front cover

real time is given in seconds. The results of these tests clearly demonstrate the usefulness (or, perhaps the necessity) for applications-specific benchmark test results when choosing a computer system. If I were in a position to purchase a system today given my needs and interests, the best choice (speed and price) would be the XXU.

A particularly interesting observation with the XXU results is that it clearly out-performs all of the other systems in intrinsic function operations. However, it yields to the IBM 3081 in arithmetic and relational operations. I could live with that. The XXU also comes in first place using the Sieve benchmark.

Another interesting note is that some of my benchmarks won't even run on the PDP 11/23 series computer which tells you something about DEC's FORTRAN compiler.

Observations and conclusions over these data are numerous. Unfortunately, I do not yet have data for the one, two, three, and four user modes on these systems similar to that previously published for the DPU and DPU with MAXIMIZER. Nevertheless, I find these results more illuminating and helpful than the accepted standard benchmark tests.

In summary, these data strongly support the choice of the XXU over other even far more expensive systems. That's great if you are in the market for buying a new system. However, here I sit using my 8MHz DPU with MAXIMIZER with a tear in my eye!

Acknowledgement: It is through the efforts of Dr. W.K. Schmidt that the data in Table 1 is available for this article. For this he has our sincere appreciation.

TABLE 1
Real Time Comparisons of Selected Computer Systems.
Run time is measured in seconds.

Benchmark	Function Type	DPU	DPU + MAX.	XPU	XXU	Mac +	IBM*** PC/AT	PDP 11/23 +	VAX 11/785	IBM 3081
Benchtest1	Single Precision Intrinsic	472	69	328	5	270	59	346	9	9
Benchtest2	Double Precision Intrinsic	1273	284	892	5	823	61	600	30	10
Benchtest3	Integer Write to File	49	45	34	8	50	82	45	11	30
Benchtest4	Integer Read from File	48	46	34	9	27	31	36	10	19
Benchtest5	Integer Arithmetic	404	98	282	13	282	289	*	xx	7
Benchtest6	Single Precision Arithmetic	1315	182	915	43	1610	297	*	26	9
Benchtest7	Double Precision Arithmetic	2995	405	2084	46	**	325	*	35	9
Benchtest8	Integer Relational	32	31	22	5	95	192	*	21	6
Benchtest9	Single Precision Relational	788	191	547	40	607	377	*	27	7
Benchtest10	Double Precision Relational	1276	380	870	41	**	410	*	44	9
Sieve		**	**	6	1	7	4	12	3	4
Number of users		1	1	1	1	1	1	1	15	90

* PDP 11 Fortran will not perform N = 1,000,000 iterations without modification
** Program not tested
xx Arithmetic overflow: program #5 aborts on VAX 11/785 and 8650 computers
Users = users logged-on during real-time tests
*** The IBM PC/AT was configured with an 8 MHz 80286 + 80287-8

CPU TIME COMPARISON (SEC)

Benchmark	XXU	VAX 11/785	VAX 8650	IBM 3081
Benchtest1	5	6	2.1	1.1
Benchtest2	5	11	3.9	1.5
Benchtest3	8	7	2.2	1.8
Benchtest4	9	7	2.1	1.5
Benchtest5	13	XX	XX	2.9
Benchtest6	42	12	4.5	4.2
Benchtest7	45	21	8.5	4.7
Benchtest8	4	7	2.3	2.2
Benchtest9	40	13	4.7	3.8
Benchtest10	40	20	5.7	3.7
Sieve	1.6	1.7	0.49	0.45

Benchmark	Purpose	Function Type	Real-Time Comparison: XXU vs. XPU
Benchtest1	Intrinsic Functions	Single Precision	66x
Benchtest2	Intrinsic Functions	Double Precision	178x
Benchtest3	Write to File	Integer	4x
Benchtest4	Read from File	Integer	4x
Benchtest5	Arithmetic	Integer	22x
Benchtest6	Arithmetic	Single Precision	21x
Benchtest7	Arithmetic	Double Precision	45x
Benchtest8	Relational	Integer	4x
Benchtest9	Relational	Single Precision	14x
Benchtest10	Relational	Double Precision	21x
Sieve of Eratosthenes			6x

BITS & BYTES

BITS & BYTES is the place to look for the odd bit of information, opinion, programs, profiles and rumors that circulate through The IACU. Our ears are always attuned to any interesting miscellany — if you have something to contribute send it along to I/O NEWS • 24843 Del Prado, Suite 473 • Dana Point, CA 92629 •(714) 661-9764

Technical Notes

Source: Cromemco NEWS, Issue #123, May/June 1987

Early XXU customers should be aware of an important engineering change. Early XXU cards were manufactured with a socket for the timing crystal. We [Cromemco] have found that, with time, the contact between the crystal leads and the socket can become intermittent. All XXU cards now have eliminated the crystal socket and instead have the crystal soldered directly to the PC board. We strongly recommend that this modification be done in the field to all of the early XXU cards. (Alternatively we will perform this modification free of charge to any XXU returned to the factory).

Besides soldering the crystal in place, the only other significant change to the XXU since introduction has been an update of the PAL ICs to assure compatibility with our ENET interface board. The current PAL ICs are marked with a part number that has a "-1" suffix. Any customer using our XXU card with our ENET interface card who has the older PAL set may return his XXU card to the factory for a PAL upgrade, free of charge.

In other technical matters, we all too frequently receive reports of system or peripheral failure due to improper system installation. Problems can occur when peripherals are tied to an earth ground (typically through the power cord) that is different in potential from that of the computer system.

This can result in an apparent sensitivity of the computer system to power line noise. If the difference in ground potential causes sufficient ground current to flow, it can also lead to the failure of serial interface drivers and receivers (typically type 1488 and type 1489 ICs) or other system failures.

Fortunately there is a solution to this ground loop problem. The solution is to **optically isolate** the computer from its peripherals. When this is done, peripherals can be installed remotely without the concern of ground currents flowing through the RS-232 serial connection. There are two different products available that can perform this job of optical isolation.

Available from **Inmac** (800-826-8180 and 800- 722-0025 in California) the **Model 8050 Fiber Optic Data Driver** offers RS-232 transmission up to 200 feet at speeds as high as 19.2 Kbaud. A pair of drivers retails for $195. Fiber optic cable is available from the same source (part number 1799) for about 60 cents per foot.

A similar product is available from **Misco** (800-631-2227). The **Model BR0963 Asynchronous short-haul modem** sells for $278 per pair, and uses readily available 22024 gauge twisted pair cable (available as Misco part number BR1805 for about 18 cents per foot). Optical isolation is included as part of the interface circuitry.

In addition to solving grounding problems, optical isolation can protect against currents induced in RS-232 lines by external electrical fields (as might be induced by a lightning storm) and other types of electrical interference. If your installation includes remote peripherals or long RS-232 lines, we strongly suggest that you contact either Inmac or Misco at the above phone numbers for more information on their optical isolation products.

Egyptian Distributorship

Source: Cromemco News Release, 6/10/87

Cromemco, Inc. announced that Kolaly Engineering has been appointed exclusive distributor for Cromemco products in Egypt. The distribution agreement was signed in Mountain View, California by Dr. Harry Garland, President of Cromemco, and Dr. Hashem El-Kolaly, Chairman of Kolaly Engineering

Dr. Hashem El-Kolaly (left) of Kolaly Engineering and Dr. Harry Garland (right) of Cromemco, Inc. at signing of Cromemco distributorship for Egypt.

In announcing the agreement Dr. Garland said, "Kolaly Engineering has an outstanding reputation for customer support and we are very proud that they will be representing us in Egypt. Cromemco has been unusually successful in international markets, with international sales now accounting for over 50% of our business. We owe our success in these markets to an outstanding team of distributors, such as Kolaly Engineering."

Kolaly Engineering is based in Cairo, Egypt and was founded in 1967 by Dr. Hashem El-Kolaly. Kolaly Engineering, with over 150 employees, specializes in the construction and computer industries, and also distributes computer terminals and printers that fully support the Arabic language. These peripherals will be integrated with the Cromemco systems and marketed by Kolaly with a full range of Arabic software.

Word Processing Added to CE

Source: Cromemco NEWS, Issue #122, April 1987

Users of Cromemco's venerable SCREEN editor have often remarked that not only is it superb for program development but, with the "beautify" command, it can also serve as a no-frills word processor. Now a word processing mode has been added to CE to bring this capability to the latest release of Cromix and UNIX.

To invoke the word processing mode, the environment variable TYPE is set to WORD (rather than to PROGRAM). If the environment variable type is set to WORD (Word Processor), the command enabled by pressing the "b" key will be BEAUTIFY. This provides text formatting (justified or unjustified) on a paragraph by paragraph, or marker to marker basis. With type set to WORD, the "p" command will advance

the cursor to the next paragraph and the "P" command will advance the screen display by one page.

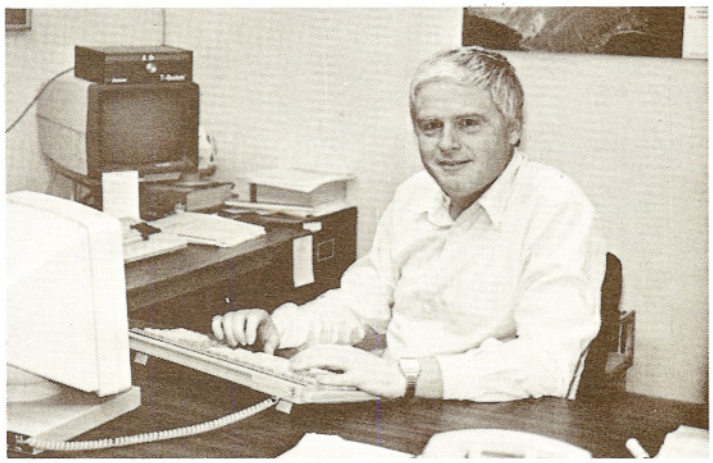
CE also now provides a facility whereby all keystrokes in an editing session can automatically be recorded in a special logfile. The keystrokes are recorded in the logfile as they are typed (without buffering). This provides for an extremely reliable method of recovering text in the event of a power failure, system failure or a catastrophic editing mistake. Using the data in this logfile, it is possible to recreate a file up to any point in an editing session. The logfile itself is an ASCII file which can be edited to provide control over the replaying of the editing session or to create special editing scripts.

The new CE can also be used interactively with Cromemco FORTRAN, Pascal and C compilers (versions 2.60 or higher) under both UNIX and Cromix-Plus. When a compiler is invoked with the **+c** option, it will in turn automatically invoke CE if any syntax errors are detected. The editor will automatically begin execution and the cursor will be located at the position in the file where the first error was detected. The editor command line will be replaced with the text of the compiler error message. Typing any key will return the editor to its normal operating mode. The text can then be changed.

Another new feature of CE is that it is able to keep track of nine separate syntax errors and assign special markers to their positions in the file. These markers can be reached via the uppercase version of the JUMP command. It is possible to move to those locations directly by specifying their numbers, or by moving to them relatively with the "next" or "previous" options to the JUMP command. When the editor displays the text on the screen, the editor command line will be replaced with the text of the associated compiler error message. After EXITing and Updating from CE, the compilation can be resumed with the CNTL-R command under Cromix-Plus or "!!" command under UNIX (C Shell).

These new features of CE are now implemented in the CE file included with the latest releases of CROMIX. The latest release of 30-series Cromix is called release XPU-151. The latest release of 40-series Cromix is called release XXU-151. This new CE will also be included with our next releases of UNIX 5.0 (for the XPU) and UNIX 5.2 (for the XXU).

These new features in CE are the result of work by Nikolaj Ivancic at Cromemco. If you have any suggestions for other CE enhancements or changes, just drop Nik a note.

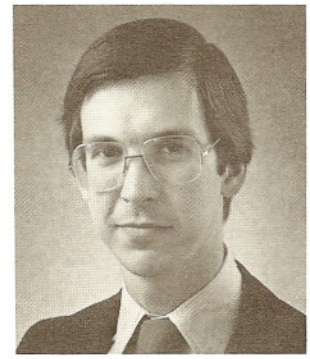


Nik Ivancic has added many new features to the CE editor, including a word processing mode.

New Cromemco VP

Source: Cromemco News Release, May 11, 1987.

Dr. Curtis Terwilliger has been appointed to the newly created position of Vice President of Engineering. Terwilliger, an 11-year veteran of the company, was formerly Director of Engineering. In his new position he will be responsible for development of Cromemco's next generation of multi-processor computers that utilize the UNIX System V operating system. He will report to Executive Vice President and Cromemco co-founder Dr. Roger Melen.



Dr. Curt Terwilliger
Vice President, Engineering
CROMEMCO, INC.

In a statement Terwilliger said, "Successful microcomputer product development is no longer solely an issue of technical innovation as it was a decade ago, nor does strict adherence to industry standards guarantee the same success as it has in the recent past. Instead, when developing the next generation of microsupercomputers one must strike a balance between conformity and creativity. The challenge for Cromemco is to optimize this balance according to the needs of the markets in which our customers exist."

In his previous position as Director of Engineering, Terwilliger was responsible for hardware and software development for the company's current line of 32-bit computers. Before that he had been responsible for the hardware development of the preceding generation of 16-bit computers and for circuit design of much of the company's original 8-bit computer line. He holds patents for his work in local area networks and phase-locked loop data separators.

Dr. Terwilliger has served as a spokesman for Cromemco and for the microcomputer industry at forums including trade shows and government briefings. He is currently serving as Chairman of the Hardware Subcommittee for the Computer System Technical Advisory Committee of the U.S. Department of Commerce. This committee advises the U.S. Department of Commerce on technology-related export matters.

Terwilliger holds a B.S.E.E. from Massachusetts Institute of Technology and M.S. and Ph.D. degrees in Electrical Engineering from Stanford University. He is a member of Eta Kappa Nu, Sigma Xi, IEEE, ACM, and AES, and a member of the M.I.T. Educational Council.

A native of Kansas, Terwilliger and his wife and two children live in Burlingame, California. He enjoys choral music, amateur radio, and hiking.

TODAY Users' Group

The most significant productivity tool for the development of database applications and the development of commercial applications software is the 4th generation language (4GL). By far the leading 4GL on Cromemco systems is TODAY developed by BBJ.

There is a new users' group for TODAY software. It is called **TUG.** The first issue of the TUG Newsletter came out in November. If you wish to join TUG, send a check payable to Adapt, Inc. for $25 ($30 outside USA) to:

TUG
P.O. Box 56
Nokomis, FL 33555
(813) 484-4686

Info Needed

68000 Lisp: We've received a number of inquiries regarding the existence of a 68000 Lisp Interpreter that will run under Cromix- Plus. So far we haven't been able to locate anything, either commercially or in the public domain (aside from XLISP). If you have, or know of such a product please contact I/O NEWS.

C-10 FIT Software: As yet another harrying April 15th came and went, an increasing demand is being voiced for C-10 software (CDOS or CP/M) to assist in the preparation of Federal

Income Tax returns. Again, we've drawn a blank. If you've developed such a program, or know of one which is available, please contact I/O NEWS. We'll pass the word around.

Curses Programming Correction

In Soup Campbell's article on Curses last issue (Vol. V, No. 6), the book "Programming with Curses" was said to be available from InfoPro Systems. IACU member Joseph Brothers learned that they no longer carry it. However, it is available from:

O'Reilly & Associates
981 Chestnut Street
Newton, MA 02164
(617) 527-4210

QUESTIONS???

Maybe we can help

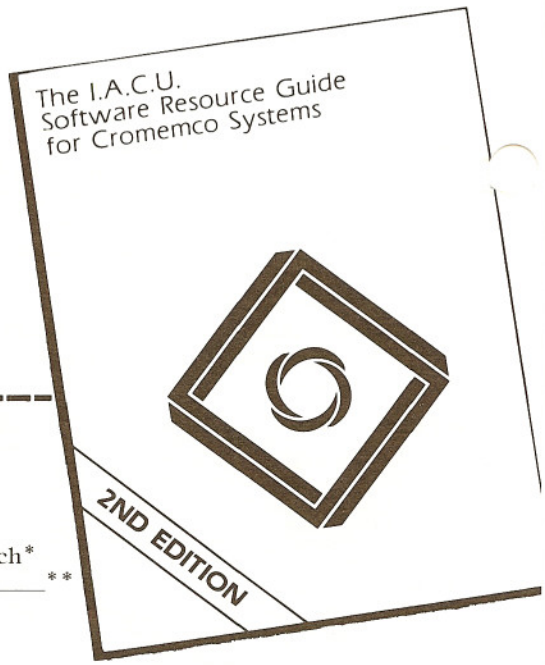
Call or write us at:

The I.A.C.U. — I/O NEWS
24843 Del Prado, Suite 473
Dana Point, CA 92629-2852
(714) 661-9764

Having problems locating software?

Now there's help ...

The IACU Software Resource Guide for Cromemco Systems



☆ Over 180 pages

☆ Multiple indices – by product name, application group, operating system, etc.

☆ Complete product descriptions — from source language to technical support

Order your copy now!
Only $50

(Shipping Charges: Over $2.50 U.S. will be invoiced. International orders must specify Air or Surface Mail and will be invoiced.)

(California residents add 6% sales tax)

The I.A.C.U. — I/O NEWS
24843 Del Prado, Suite 473
Dana Point, CA 92629-2852

Send __ copies @ $50.00 each*
I enclose $_____**
☐ Check ☐ Money Order
(make payable to The IACU)

Charge my ☐ MasterCard ☐ Visa
Phone orders with credit card call (714) 661-9764

Card No._____ Expires_____

Signature_____

(Please Print)
Name _____

Address _____

_____

City _____ State _____ Zip _____ Phone _____

Country_____

* Calif. residents add 6% sales tax
** U.S. Dollars. International checks not drawn on U.S. bank add $10

New Products

Continued from page 18

UNIFOCUS

UNIFOCUS is a unique interface to the UNIFY DBMS from LEVEL II MICROFOCUS COBOL. UNIFOCUS supports 55 COBOL-callable functions which provide applications developers complete flexibility to manipulate the database. Functions are provided to add, delete, modify/update, sequential or keyed access of records, buffered access, search of B-tree index, field checks and almost all other functions supported by the C Language interface to UNIFY.

Flexibility

UNIFOCUS is designed to provide a high degree of flexibility whereby addition, deletion or modification of fields from a record type do not require re-compilation of the COBOL program as long as they are not being actually used by the application program.

With UNIFOCUS, developers are able to solve problems they haven't been able to solve before, to create applications which run faster and with better end-user response time, and to have more control over functions of the DBMS. UNIFOCUS is also structured to provide ultimate flexibility for developers.

Speed

UNIFOCUS is the absolute fastest method of transaction processing by an application. Instead of processing database requests through an SQL interpreter, UNIFOCUS processes them using compiled C subroutines embedded in MF-COBOL, running as fast as the rest of the program.

Enhancements

UNIFOCUS provides exactly the same functions as the RM-COBOL interface provides. Users already familiar with the RM-COBOL interface have only to know of the syntactical difference between the compilers to use UNIFOCUS and exploit the speed advantage of Microfocus COBOL over RM-COBOL. In addition, this interface has enhanced the capabilities provided with the RM-COBOL interface with UNIFY.

The limitation of the RM-COBOL interface in using COMBined field types of the database are removed in UNIFOCUS. Whereas RM-COBOL prohibits the use of COMB fields in COBOL programs using the interface, and hence one cannot access COMB fields which are referenced, UNIFOCUS allows the use of COMB fields and even provides accessibility of COMB and REF fields. Thus, if there is a COMB field in the record, one can access all of the sub-fields by either giving the COMB field number in the field list array and define the view for all sub-fields, or by giving all sub-fields in the field list array.

For further information on UNIFOCUS, contact:

Marketing Department
Zenith Computers Limited
Zenith House
Cardinal Gracias Road
Andheri (East)
BOMBAY 400 093
INDIA

IT'S HERE!
dBIII Compiler
for UniPlus + System V

Software Standards is now shipping dBIIICompiler from Word-Tech Systems for Cromemco's UniPlus System V machines. dBIIICompiler is a powerful dBase III compiler as well as a full-function program development system. No other software or product (such as dBASE III) is required to use dBIIICompiler although it is compatible with dBase II and dBase III files from any other machine. With dBIIICompiler under UniPlus System V, you can quckly develop or port anything from small utilities to giant multi-user data processing systems. In addition to all the features found in our Cromix-Plus version, including multi-user record level locking, terminal independence and dBase II data file compatibility, our new UniPlus version generates memory-efficient **SHARED CODE COMPILED PROGRAMS** and is 100% source code compatible with our Cromix-Plus version. Our latest release also incorporates **MULTIPLE PRINTER** support and complete control over the spooler. Suggested list price is $995 and there are no royalties for distributing compiler programs.

RealWorld Accounting Packages

RealWorld is a full-featured accounting system which includes A/P, A/R, G/L, Payroll, Inventory, Order Entry, Billing, and Sales Analysis. All modules are fully integrated and the system is multi-user. RealWorld is available on both Z80 and 68000 Cromix as well as UniPlus System V. Suggested list price is $695 for Z80 versions and $795 for 68000 Cromix and UniPlus versions.

Contact Software Standards or your Cromemco Dealer for more information. Dealers can request Discount Schedules, Literature and Demo Packages.

Software Standards, Inc.
Box 3181, Lake Charles, LA 70602-3181
(318) 433-3690

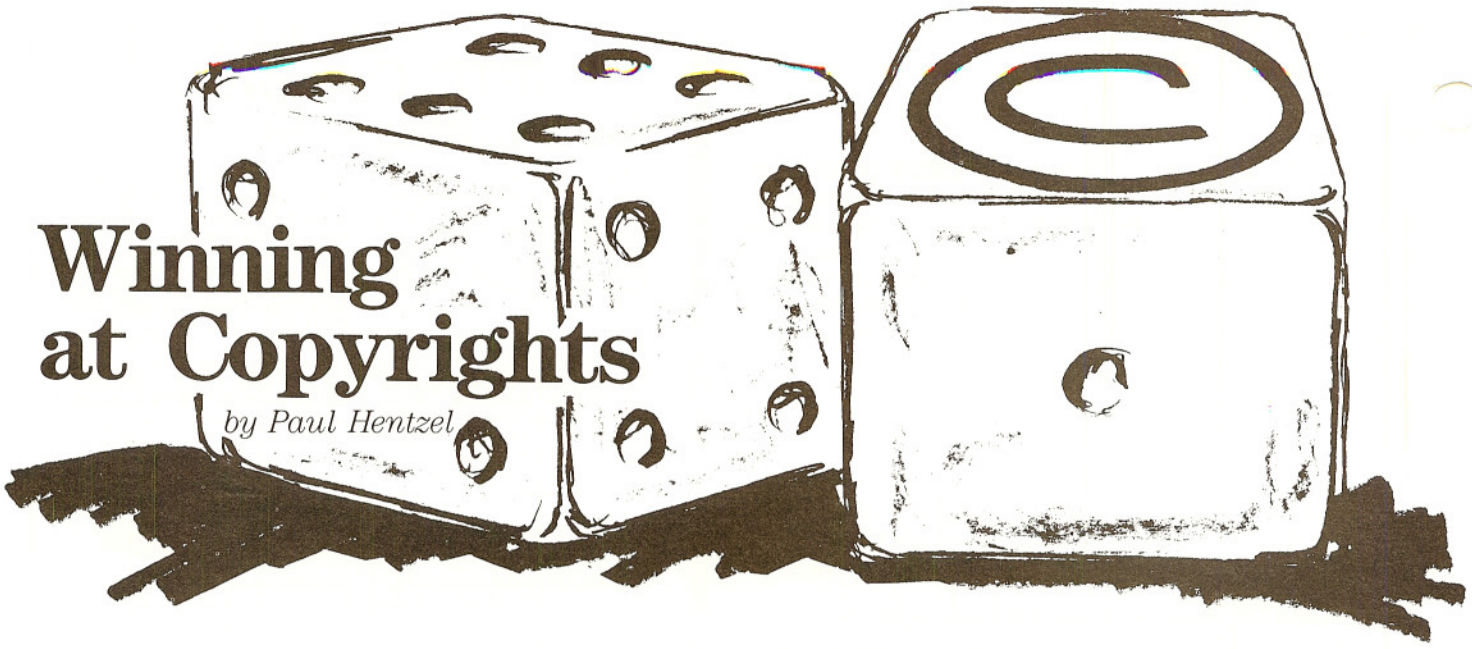
Trademarks: UniPlus: Unisoft Systems; dBase II and III: Ashton-Tate; Cromix-Plus: Cromemco, Inc.; dBIIICompiler: Wordtech Systems; RealWorld: RealWorld Corp

GO WITH THE SPECIALISTS IN CROMEMCO FOR OVER 8 YEARS

2 Megabyte Ram Card with Parity for DPU/XPU/XXU for Cromix & UNIX Systems	**$1795**
IPC Intelligent I/O Controller Z80 8 Mhz. Processor, 32K two-port Ram I/O Buffer, 0-38K & autobaud supported. Cromix (and soon UNIX) drivers included.	**$1095**
Above boards come with 1 year warranty and 30 day money back guarantee.	
OCTART Upgrade from 3.6864 Mhz. to 4.5/5.0/5.5/6.0 Mhz. Operations. Increase throughput by up to 65% (Speedup varies from board to board).	**$ 125**
OCTART Drivers for Cromix (and soon UNIX). Increase throughput by 300% or more.	**$ 195**
DPU Upgrade to 68010 @ 10Mhz. (not for WDI systems)	**$ 225**
256KZ Upgrade to 1024KZ	**$ 650**
SPTYP: Cromix Spool Utility translates WriteMaster print codes to your printer based on printcaps file	**$ 95**

Prices subject to change without notice
Dealer Inquiries Welcome

Microcomputer Specialists Inc.
P.O. Box 88127
Grand Rapids, MI 49518
(616)942-5412 or (616)776-9646



Winning at Copyrights

by Paul Hentzel

Placing a Copyright Notice on Your Software

The first and most fundamental step for you to take towards protecting your copyright interest in software (and all related Works) is to place a copyright notice directly on all copies of your software (and on every copyrightable Work that relates to your software). This copyright notice must include:

1. The circle C symbol © (or the word Copyright),

2. The year of first publication, and,

3. Your name as an individual owner of the copyright interest (or your corporation's name).

The notice must be visible (not necessarily prominent) during normal use. The notice should be separate from any other printed material on the Work such as addresses, publication numbers, and credits. The following notice formats are frequently used:

© Copyright 198? Your Name? Inc?
Copyright 198? Your Name? Inc?
© 198? Your Name? Inc?

Each of these notices have all three of the mandatory elements of a valid copyright notice. The word "Inc" or "Corporation" may be omitted in a tight space situation; if other text on the same page (or label) fully identifies the owner. The word "Copyright" and the circle C symbol © are redundant; and only one is required. Therefore, the full word may also be omitted to conserve space.

The word "copyright" only has significance in English speaking countries. Whereas the circle C symbol is recognized throughout the industrialized world under the Universal Copyright Convention (UCC). Merely by placing the circle C type notice on your software, you preserve your copyright interest around the world in all of the UCC member countries.

Using both the word and the circle C symbol in a notice does have one advantage peculiar to the US. If your publisher does not print the circle C symbol properly, the presence of the full word "Copyright" may save the notice and preserve your copyright interest. The circle C symbol is not a standard text character, and is subject to what the Copyright Office refers to as "deviate variations" of the notice requirement. Common deviates are no circle, partial circle formed by opposed parenthesis (C), and a C within a hexagon (dot matrix character). Use of deviates may result in a defective notice, and possible loss of your copyright interest. My advice is to avoid all deviates, and be ever vigilant against the above common deviates; and report them to your local Patent/Trademark attorney at once.

The phrase "All Rights Reserved" is commonly used on a second line of the notice. This phrase concerns the Buenos Aires Convention, and is required to obtain protection in Bolivia, Honduras, and Uruguay. The phrase is of little significance in most commercial situations.

Label Notices

The copyright notice for software has several possible locations. A gummed label on the floppy, cassette, cartridge, or other immediate housing is the traditional and most prominent location. The copyright notice is typically placed along the bottom edge of the label spaced from other text such as the software title, version number, and manufacturer information. If the dust cover has a label, then place a notice there as well. One notice is sufficient; but multiple notices "can't hurt." Your primary notice might get lost in the communication with your print shop, or get covered by another label, or it might even come off.

If your software is an operational program in a ROM, place a tiny label and notice on top of the ROM. The notice must be visible under ordinary examination; however, people expect to squint a little when reading ROM tops. A secondary notice site for your operational programs is the circuit board (front or back) along the edge. Both the ROM and the board notice are considered visible during normal use even though they are normally mounted under a housing.

The carton or trade box for your floppy or circuit board is a less effective location for your copyright notice because the box may not be around during ordinary use. The board is in the consumer's computer and the floppy is in his drive. The box is not normally involved in the use of your software. However, placing a notice on your carton is easy to do and will also protect any copyrightable Works on the container (trade dress).

Printout Notices

Your copyright notice must appear on all printouts of your software (preferably near the beginning or end); whether the printout is dumped by the user from his machine or formally published in the computer literature. Printouts are not closely related to or-

dinary use; but they must be submitted to the Copyright Office along with your application. If the printout does not have a notice, the Examiner will request an explanation (major hassle and possible loss of rights). The printout submission must be the "best edition" (latest version), and will identify the Work which is protected by your copyright. I will cover the best edition problem in a later column.

The notice portion of the printout must be in human eye-readable form to establish with the Examiner that your software actually included a copyright notice. The Examiners cannot read machine language, so this small portion should not be in hexadecimal or source code. The circle C symbol is not a standard text character and creates a problem in the printout notices. The closest available symbols are the "deviate variations" (C) and C within a hexagon. It is unclear whether the US Copyright Office (or any other copyright office) will accept these variations on computer printouts. The UCC specifically requires an actual circle C. You can use the full word "Copyright" for full US protection; but that does not satisfy the UCC either. So, use either the deviate form alone or with the full word "Copyright." It is the best that you can do in this "Catch 22" situation.

Display Notices

Your copyright notice for application software must appear on the monitor display and be visible to the user during ordinary use. The display notice is usually formed by the same code as the notice that appears in your printout of the best edition. Your display notice may be a single event (prolonged) during the program introduction; or the notice may be displayed continuously throughout the program. If your application software has several user sections, you should place a notice at the start of each section.

Related Works

The manual, instruction sheet, flowchart, diagram, keyboard template, or other printed material accompanying your software each have potential as a copyrightable work. You may place a copyright notice on all of these related items. Such a notice would be visible during ordinary use, and may have back-up possibility if the code notice in your program fails. The looseleaf format brings up special considerations. The pages are removable and will require a copyright notice on each page.

In addition, the year portion of the notice functions as a revision flag for annual updates.

If you have any questions or suggestions for future articles you may contact me directly at:

Paul Hentzel
Patent Trademarks Copyrights
441 Nevada Avenue
Palo Alto CA 94301
(415) 326-8254 9am-midnight

Editor's Note: *Paul Hentzel has been advising Cromemco, Inc. on copyright matters since 1976. The next installment will cover the preparation of software copyright applications.* CD

FOR SALE

System I—W/DPU, 256KZ, 64FDC, TUART, WDI, 20Mb hd, DSDD 5¼"FDD—$5500.

System III-W/ZPU, 192 K, 16FDC, TUART, WDI, 5Mhd, DSDD 8" FDD, DSDD, Persci—$3000.

3355A Printer w/tractor, cables & PRI—$2500.

As priced or best offer—(704) 264-3437

CHANGE OF ADDRESS

Membership # _____
Type or Print
Your Name _____
New Address
_____
_____
New area code & phone No. (___) _____
Sign Here _____
Date new address in effect _____

Send address changes to I/O News,

THE BEST OF C-10 SOFTWARE

Applied Environmetrics now distributes **TYPEQUICK**, the best available typing tutor for the C-10. **TYPEQUICK** teaches the fundamentals of touch typing in ten personalized lessons.

Other fine software products supplied by ***Applied Environmetrics*** are the **C-10 FUN DISK** as well as eight volumes of **THE BEST OF PUBLIC DOMAIN SOFTWARE.**

TYPEQUICK $ 69.50 **THE C-10 FUN DISK $100.00**

THE BEST OF PUBLIC DOMAIN SOFTWARE ($25.00 each)

Volume 1 — Games
(Collosal Cave Adventure, Sbasic Startrek, Trade, etc.)
Volume 2 — Communications
(Modem10, SQ — File squeezer and unsqueezer, etc.)
Volume 3 — Utilities
(Undelete, Catalogue, LU library utility, etc.)
Volume 4 — Assemblers
(8080 and Z80 assemblers and a loader)
Volume 5 — Disassemblers
(Z80 Disassembler and a Tracer)
Volume 6 — Basic
(Tiny Basic, Sbasic Nolist cracker)
Volume 7 — Chess
(Fortran Source and Executable Code)
Volume 8 — Geneaology
(dBASE II program to enable you to trace a family tree)

Source code is supplied on most of the Public Domain Volumes. They can therefore be easily altered for particular applications and offer valuable examples in applications programming. All prices are in US Dollars and include air mail delivery. Further details from ***Applied Environmetrics, (03) 817-2571.***

ORDER FORM for THE BEST OF C-10 SOFTWARE

Applied Environmetrics • **118 Gordon Street** • **Balwyn, Vic. 3103** • **Australia**

PUBLIC DOMAIN SOFTWARE is $25.00 per Volume
Please send:

__ copies of Volume 1 (Games) $ ___
__ copies of Volume 2 (Communications) $ ___
__ copies of Volume 3 (Utilities) $ ___
__ copies of Volume 4 (Assembler) $ ___
__ copies of Volume 5 (Diassembler) $ ___
__ copies of Volume 6 (Basic) $ ___
__ copies of Volume 7 (Chess) $ ___
__ copies of Volume 8 (Geneaology) $ ___
__ copies of TYPEQUICK @ $69.50 $ ___
__ copies of the FUN DISK @ $100.00 $ ___
Total Amount Enclosed $ ___

Name _____
Street & No. _____
_____
City _____
Zip Code _____ Country _____

Payment to be by check or bank draft in US Dollars drawn on a United States bank, or in Australian Dollars drawn on an Australian bank.

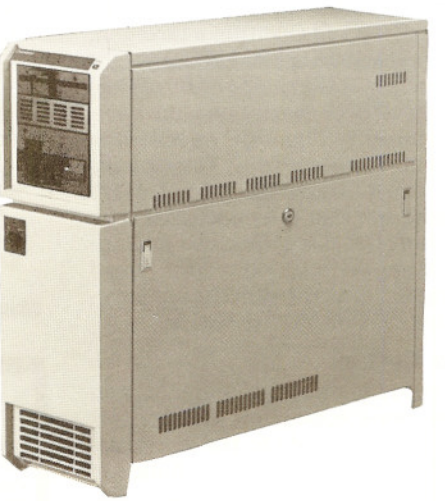
BEST IN SALES AND SERVICE

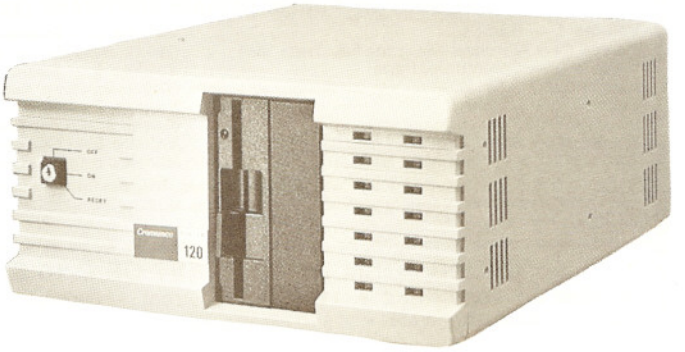
If you're looking for a local Cromemco dealer, search no more. **Computer Crossroads** is here, and ready to help you with the most qualified Cromemco sales and service team in the U.S.

Based in the greater Dallas area, we sell and service Cromemco equipment in all regions of the U.S. and in several foreign countries. Our central location and convenient access to DFW Airport means equipment can be received from any location in the U.S. and returned to the customer within three working days (excluding transit time), when our expedited service is requested.

On top of that, you get the kind of expertise and support that has earned **Computer Crossroads** a host of Cromemco's "Outstanding Service" awards and the one and only "Number 1 U.S. Dealer" award. When it comes to quality, we can't be beat.

Service and support—more than just words—they are the reasons behind our phenomenal success. Simply said, sales backed by support means success and that means **Computer Crossroads**.





COMPUTER CROSSROADS of AMERICA, INC.

SOFTWARE
SERVICE
HARDWARE

Where It All Comes Together

Superb pricing on Cromemco equipment with the best in support at no extra cost.

1750 Alma Road • Suite 118 • Richardson, Texas 75081

FOR SUPER SERVICE CALL (214)231-6108 Telex: 4991118

FAST•FAST•FAST•

Now! Upgrade Your Cromix or Unix System to

1,050,000

WHETSTONES PER SECOND

FASTEST CPU RATE YET

Now you can soar to the outer limits of modern software capability at processing rates never before possible. Upgrade your current Cromemco to the speed of our newest systems with the remarkable XXU processor board. It combines the proven Motorola 68020 chip and the Motorola 68881 co-processor into a single unit with integral high speed cache memory -- larger and faster than any other computer in its class. These figures tell the story.*

MANUFACTURER	MODEL	WHETSTONES/SEC.
CROMEMCO	**XXU Based**	**1,050,000**
DEC	MICROVAX II	887,000
SUN	3/50	860,000
APOLLO	3000	780,000
DEC	VAX 11/780	476,000
IBM	PC/RT	200,000

*Test data obtained from Datamation and Unix World magazines.

LOW COST

As little as $4995 will convert your recent model Cromemco system to this high speed capability. Older models will cost somewhat more, but still less than half the cost of a new system. Low purchase price, however, is only the tip of the iceberg. The big cost savings come into play once you've started using your XXU-based system.

EASY UPGRADING

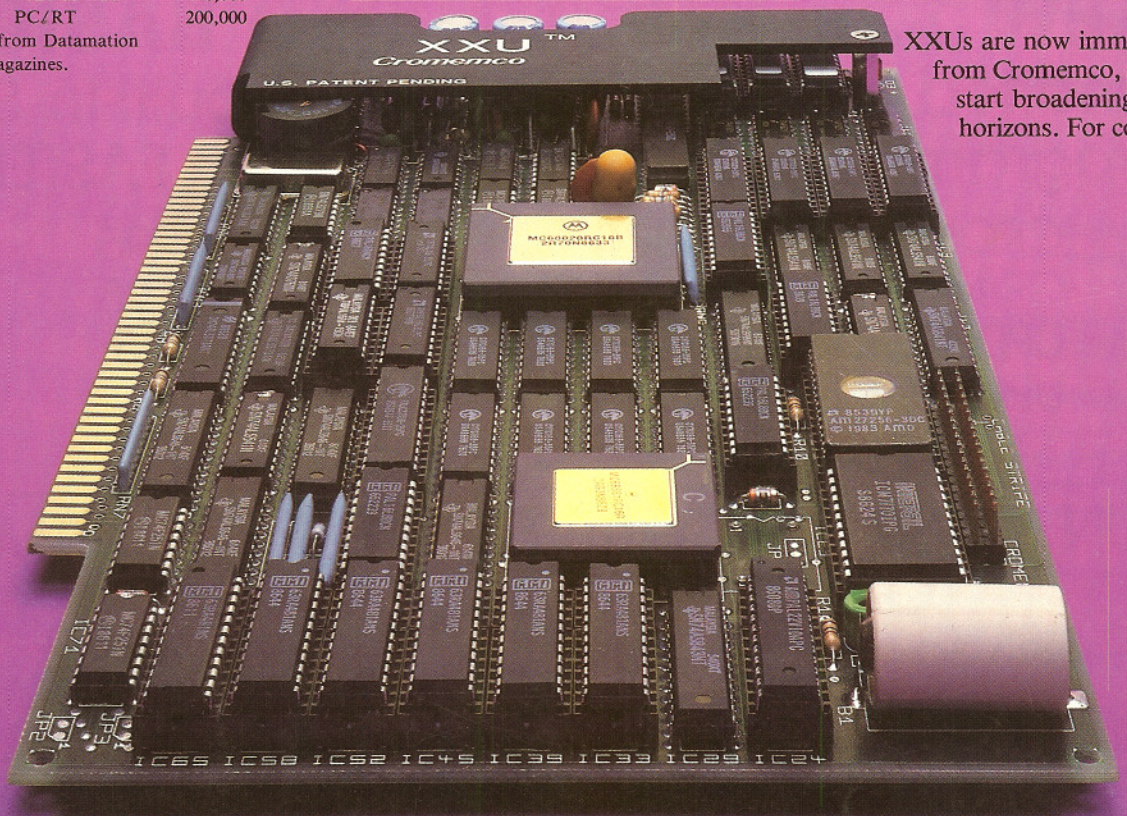
Depending on your model of Cromix or Unix system, upgrading may be as simple as plugging in a single board. Older systems can be easily upgraded by your dealer or at the factory. Either way, your new XXU is fully backed by Cromix and Unix software support and our factory warranty.

PLUS THESE ADDED CONVENIENCE FEATURES

A real time clock/calendar, powered by a seven year lithium cell battery, eliminates time and date setting tasks. The on-board XDOS/boot ROM program perfor system diagnostics and controls a fault-detect LED on the XXU board.

XXUs are now immediately available from Cromemco, so order now and start broadening your processing horizons. For complete data, call or write today.

XXU™
Cromemco
U.S. PATENT PENDING



Cromemco
A DYNATECH COMPANY

CROMEMCO, INC.
280 Bernardo Ave., P.O. Box 7400
Mountain View, CA 94039
(415) 964-7400

IN EUROPE: Cromemco GmbH
6236 Eschborn 1, Frankfurter Str. 33-35
P.O. Box 5267, Germany
0049 (06196) 481606